# QlikView

# Server/Publisher

*Version 11.20 SR7 for Microsoft Windows®*

*Lund, Sweden, 2014*

*Authored by QlikTech International AB*

# Contents

# Part 1 Introduction

# 1 Overview

This document describes QlikView Server and contains information on installation, architecture, security, and licensing. The document also includes a number of appendixes that provide additional in-depth information.

## 1.1 QlikView

### QlikView Server

QlikView Server is a platform for hosting and sharing QlikView information over an intranet or the Internet. QlikView Server connects users, client types, documents, and objects within a secure environment.

### QlikView Publisher

QlikView Publisher manages content, access, and distribution. By reducing data, each user can be presented with tailored information. The QlikView Publisher service and user interface are fully integrated into QlikView Server and QlikView Management Console (QMC).

## 1.2 QlikTech Support Services

Contact QlikTech if product support, additional training, or consultation concerning application development is needed. Consult the QlikTech homepage for current information on how to get in touch with the support services:

**http://www.qlikview.com**

QlikTech International headquarters:

**QlikTech International**
**150 N. Radnor Chester Road**
**Suite E220**
**Radnor, PA 19087**
**USA**

**Phone: +1 (888)-828-9768**
**Fax: 610-975-5987**

For other locations, visit the QlikTech home page (see above).

## 1.3 Conventions

### Style Coding

Menu commands and dialog options are written in **bold**. File names, paths, and sample code are written in `Courier`.

## Environment Variables

The paths described in this document use environment variables. The variables and the equivalent paths in Windows Vista® (and later) and Windows XP are presented below.

| Environment Variable | Windows Vista and later | Windows XP |
|---|---|---|
| %ProgramData% | C:\ProgramData | C:\Documents and Settings\All Users\Application Data |
| %ProgramFiles% | C:\Program Files | C:\Program Files |
| %UserProfile% | C:\Users\[username] | C:\Documents and Settings\[username] |

# 1.4 About this Document

This document describes QlikView Server and QlikView Publisher version 11.20. The contents of the software as well as the document may change without prior notice.

# 2 What's New in QlikView 11 Server?

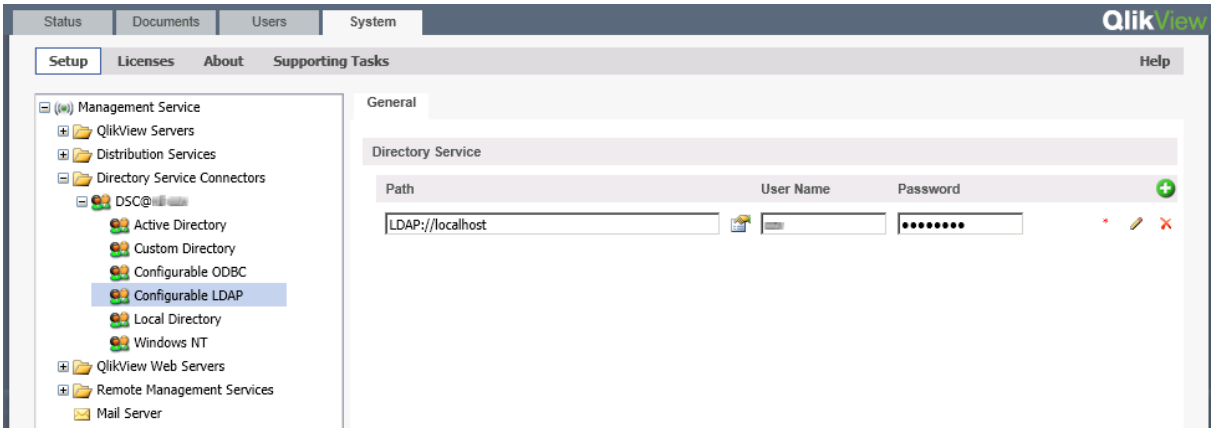
This chapter describes the functionality that has been added or improved in QlikView 11 Server.

## Context Sensitive Help

Context sensitive help has been added in QlikView Management Console (QMC).


## LDAPs

Configurable LDAP DSP for LDAPs (LDAP via SSL) support has been added.

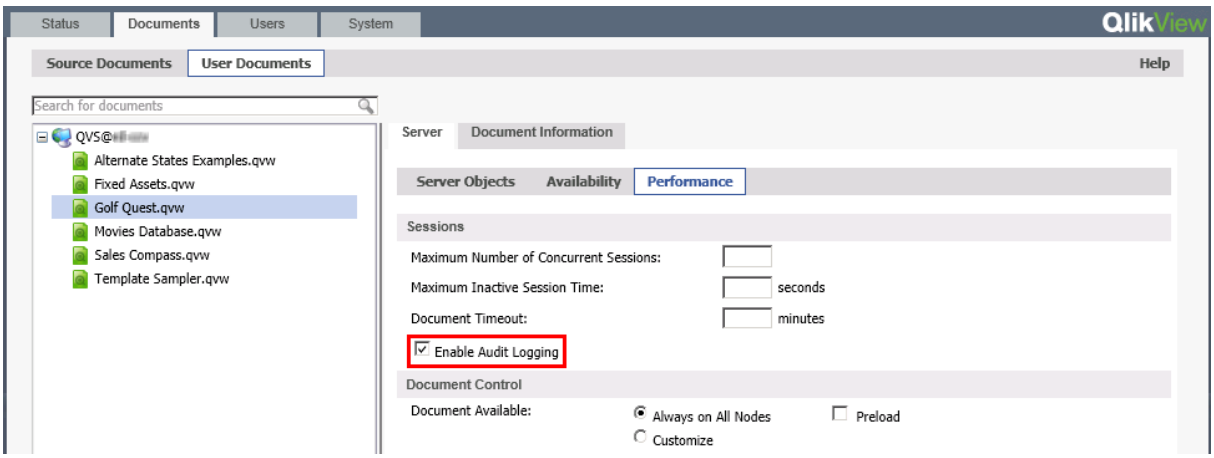



## Audit Logging by Document

In some cases, it is required to generate a user audit log, so that every query is registered and it is possible to see "who did what" retroactively.

QlikView 10 Server can do this; however, the switch is for the entire server – either all documents are logged or no document is logged. In addition, if there is a large number of selections in a list box, not every selection is logged.

In QlikView 11 Server, this logging can be done per individual document. In addition, logging of every selection can be enabled.

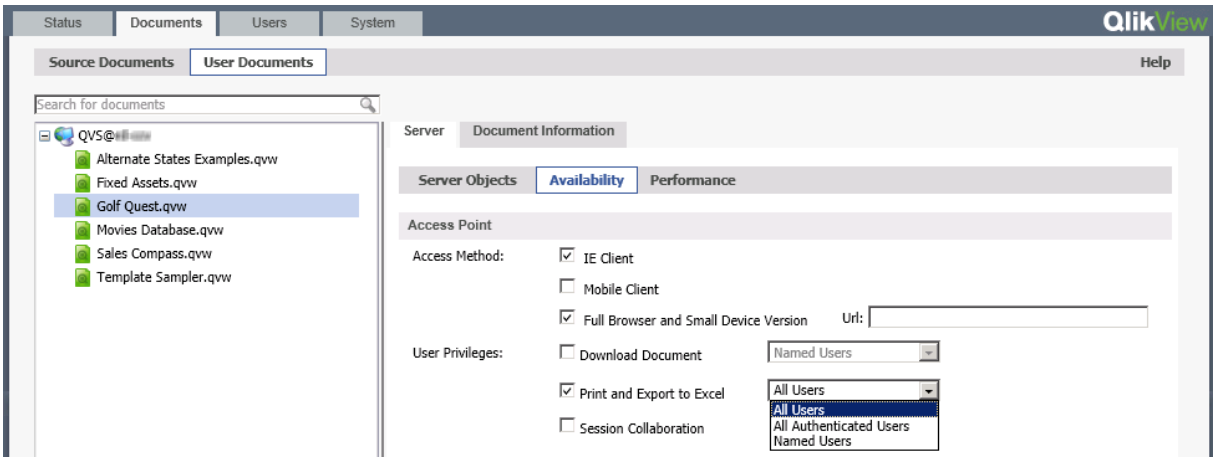



## Enable/Disable Document Download, Exporting, and Printing per Document and User

In many situations, the system manager wants to prevent any "hard" data coming out of QlikView Server. In these situations, downloads of the `.qvw` file, printing, and exporting are not allowed; only the interactive session with QlikView Server is permitted.

In QlikView 10 Server, this functionality is available at the document level only for downloading.

In QlikView 11 Server, this has been improved so that the functionality is available on a per user level as well as the capability to enable and disable exporting and printing on a per document and user level.




## Supporting Task for .qvd Creation

The creation of `.qvd` files can be added as a Supporting Task.

**Note!** This is *not* a replacement for creating `.qvd` files using a `.qvw`. See the QMC online help for more information.

## Distribution to Email within a .qvw Document

A `.qvw` file can be distributed to email recipients defined in a field in the document.

## Alert Email to Document Administrators

Alert emails can be sent to document administrators.

# License Tracking

The use of licenses has been added to the QlikView Event Server logs. The following events are now logged (when using low verbosity logging):

- "PGO", "Recreating [file name] from backup"
- "PGO", "Recreating corrupt file [file name]"
- "PGO", "Creating file [file name]"
- "License", "License leased to user [user name] on machine [machine name]"
- "CAL usage", "Using CAL of type [CAL type] for user [user name] on machine [machine name]. Sessions on this caltype: X"
- "CAL usage", "Releasing session CAL for user [user name] on machine [machine name]"
- "CAL usage", "Usage CAL session for user [user name] on machine [machine name] stopped"
- "CAL usage", "Named User CAL session for user [user name] on machine [machine name] stopped"
- "CAL usage", "Document CAL session for user [user name] on machine [machine name] stopped"
- "CAL allocation", "Unused (Document) Named User CAL [user name] deleted – ok"
- "CAL allocation", "(Document) Named User CAL (not used for 24 hours) [user name] deleted – ok"
- "CAL allocation", "Unused (Document) Named User CAL [user name] marked for deletion – ok"
- "CAL allocation", "(Document) Named User CAL [user name] added – ok"
- "CAL allocation", "Named User CAL (not used for 24 hours) [user name] deleted – ok"
- "CAL deallocation", "(Document) Named User CAL [user name] no longer marked for deletion – ok"
- "CAL deallocation", "(Document) Named User CAL [user name] not marked for deletion – denied"
- "CAL deallocation", "(Document) Named User CAL [user name] not found – denied"

# Distribution and Reload Performance

The performance of reload and distribution has been improved.

## Reloads

Prior to QlikView 11 Server, a reload is performed in the following way with Publisher:

1. The entire document (`.qvw`) is loaded to memory from disk.
2. A reload is completed.

In QlikView 11 Server, a reload is performed in the following way:

1. The document (`.qvw`) without the data is loaded to memory from disk.
2. A reload is completed.

The performance enhancement is the reduction in time to load the document to memory from disk, since there is no data. QlikView 11 Publisher can open source documents without data prior to executing a reload task. There is no need to load the document data to memory and then perform a reload of the document.

## Loop and Reduce

Prior to QlikView 11 Server, a loop and reduce is performed in the following way with Publisher:

1. The entire document is loaded to memory from disk.
2. The document is reduced and saved to disk.
3. Go to Step 1 until the Loop is completed.

In QlikView 11 Server, a loop and reduce is performed in the following way:

1. The entire document is loaded to memory.
2. The document is reduced while being duplicated in memory.
3. The document is reduced and saved to disk.
4. Go to Step 2 until the Loop is completed.

The performance enhancement is the number of times the document is loaded from disk for each loop. However, the memory footprint is increased (based on the largest slice during the loop and reduce) for the document.

## QlikView Management Console User Interface

The QMC user interface has been improved:

- User interface inconsistencies have been cleaned up.
- The performance of refreshing tables all over the QMC has been improved.
- The Status page is drawn and updated faster.
- All of a task chain can be expanded by right-clicking.
- A user can be removed from all distributions where the user is explicitly targeted.
- Search and filters have been added to Source Documents, User Documents, and Tasks.
- Alerts within QMC have been added for service status.
- Clustering and user types have been made more consistent among the services.

## Reduction with Lock Fields

In previous versions of QlikView, a reduction was affected by lock fields (by in effect reduce on the locked values). In QlikView 11 Server, the reduction ignores any locked fields.

## Improved Logging

Changes to the settings in QlikView Server and QlikView Web Server are stored in the audit log.

The logging and error handling have been improved for QlikView Distribution Service.

## QMC and QMEC are Merged into QMC

QMC has been removed and QlikView Enterprise Management Console (QEMC) has been renamed to QMC.

## AccessPoint Remake

AccessPoint has been given a new look and feel:

- New search capabilities have been added.
- The document attributes have been leveraged to improve the categorization of documents.
- Document descriptions can be added.
- Global messages can be shown on the AccessPoint by adding messages in QMC.

## EDX Enhancements

Starting an EDX returns a session ID to allow interrogation of the status of the session rather than on the task. When the session is done, the status contains a list of all the tasks (and session IDs) that have been triggered, allowing for continuous tracking of the status. This functionality is available through the API.

## Load Balancing Improvements

A new algorithm, "CPU with RAM Overload", for load balancing when using a QlikView Web Server has been added for improved management of a cluster of web servers. In essence, the web server can now route traffic based on RAM and CPU use.

## Retries

If a task that contains a loop fails, it restarts from the point of failure, rather than from the beginning of the loop.

## MSI – Installation of QlikView Server

The usability of the MSI has been improved.

## QlikView Settings Service

When Microsoft® IIS is used as web server, a new support service, QlikView Settings Service, allows IIS to be managed via the same port (4750) that is used to manage QlikView Web Server.

# Part 2 Installation

# 3 Upgrading QlikView

## 3.1 Upgrade Considerations

### Migrating from 32-bit (x86) to 64-bit (x64) Version

When migrating from 32-bit (x86) to 64-bit (x64) or from 64-bit (x64) to 32-bit (x86), the running version *must* be manually removed prior to installing the new version.

### Upgrading from Version 8 to 11

Upgrading from QlikView Server version 8 to version 11 requires upgrading to version 9 prior to upgrading to version 11.

**Note!** QlikView Server *cannot* be upgraded directly from version 8 to version 11.

### Upgrading from Version 9 or 10 to 11

Considerations for upgrading from QlikView version 9 or 10 to version 11:

- The installation of QlikView Server requires a reboot of the machine for proper operation.
- QlikView Server version 9 handles EDX triggers via HTTP POST calls only. In QlikView Server version 10 and 11, EDX triggers are triggered by the QlikView Management Service (QMS) API, where more granular functionality is available. See the QMS API documentation in the Management Consoles and the QlikView SDK for usage instructions.
- QlikView AccessPoint is the default start page for QlikView Server.
- Previous Management Consoles for QlikView Server and Publisher have been completely replaced by QlikView Management Console (QMC). The QMC must be started to register a license for QlikView Server, unless a valid license is already available on the machine running QlikView Server.
- Anti-aliasing on fonts is no longer available.
- QlikView has a common file format for versions 7, 8, 9, 10, and 11.
- Windows 2000® is no longer an officially supported host operating system.
- See the Release Notes for more information on upgrading to QlikView 11.

## 3.2 Upgrade Procedure

For a successful upgrade of QlikView Server, take the following basic practices into account:

- Back up the current QlikView data directory, which includes most of the log and some of the configuration files as well as the document folders. The files are typically located in the following location:
  Windows 7 and later, Windows Server 2008 and later: `%ProgramData%\QlikTech`
  Windows XP, Windows Server 2003: `C:\Documents and Settings\All Users\Application Data\QlikTech`
- Perform the upgrade during a scheduled downtime – QlikView Server must be stopped for the upgrade to be successful.
- Licensing information and settings are saved by default when QlikView Server is removed. They are applied to any subsequent installation of QlikView Server on the system.

**Note!** The installation does not support upgrade from beta or release candidate versions of QlikView 11 Server.

To install QlikView Server, proceed as follows:

1. Verify that backup media exists for the current release of QlikView Server and back up all current files associated with QlikView Server (HTML pages, QlikView documents, licensing file, QlikView Server `.shared` files, and so on).
2. When running QlikView Server version 8, use the **Users** tab in QMC to determine if there are any active users linked to QlikView Server. It may be a good idea to send out a broadcast message to notify the users that the service will be stopped.
3. Uninstall QlikView Server from **Start Menu>Control Panel**.
4. Install QlikView 11 Server.

**Note!** When upgrading from a previous version and using Microsoft IIS, the virtual folders in IIS must be updated (see the table below).

| Microsoft IIS Virtual Folder | Update Required |
|---|---|
| `QVAJAXZfc` | Update to `%ProgramFiles%\QlikView\Server\QlikViewClients\QlikViewAjax`. |
| `QvPlugin` | Update to `%ProgramFiles%\QlikView\Server\QlikViewClients\QlikViewPlugin`. |
| `QvClients` | Update to `%ProgramFiles%\QlikView\Server\QlikViewClients`. |
| `QvAnalyzer` | Removed |
| `QvJava` | Removed |
| `QvPrint` | Removed |

## 3.3 Multi-machine Preparation

When upgrading an installation that is spread over multiple machines, extra planning is required, since versions cannot be mixed arbitrarily.

### Simple Upgrade

This procedure requires no special planning and involves the smallest risk, but causes the system to be down for some time.

Proceed as follows to perform a straight-forward upgrade:

1. Perform a backup as described in *Upgrade Procedure (page 19)*.
2. Stop all services running on all machines.
3. Upgrade the services on each machine (in any order).
4. Start all services on all machines.

### Maximize Uptime

This procedure requires more planning, but the system uptime (from an end user point of view) is maximized.

Proceed as follows to perform the upgrade:

1. Perform a backup as described in *Upgrade Procedure (page 19)*.
2. Stop QMS (which means QMC becomes unavailable).
3. Upgrade in the following order (let the installer restart the services):
   a) Web servers
   b) Directory Service Connector (DSC)
   c) QlikView Server (QVS)

d) QlikView Distribution Service (QDS)
e) QMS

4. Start QMS (which means QMC becomes available again).

# Migration to a New Machine

An alternative way is to build the new environment on new servers.

**Note!** When upgrading from a previous version and using Microsoft IIS, the virtual folders in IIS must be updated (see *Upgrade Procedure (page 19)*).

Proceed as follows to perform a migration to a new machine:

1. On the new machine, install a running, licensed version of QlikView 11 Server.
2. Stop all QlikView services on the old machine.
3. Remove or rename the `%ProgramData%\QlikTech\ManagementService\QVPR` folder.
4. Remove or rename the `%ProgramData%\QlikTech\ManagementService\qvpr_<NewMachineName>.ini` file.
5. Copy the `QVPR` folder and the `.ini` file "as is" from the old machine to the new one (that is, keep the folder name):
   Version 9: `%ProgramData%\QlikTech\Publisher\CommandCenter`
   Version 10: `%ProgramData%\QlikTech\ManagementService`
6. Rename the `.ini` file (that is, change `qvpr_<OldMachineName>.ini` to `qvpr_<NewMachineName>.ini`).
7. Change all references to `<OldMachineName>` to `<NewMachineName>` in the QVPR `.xml` files.
8. Start the QlikView services on the new machine.
9. In QMC, change the source folder path to the correct folder (or the tasks cannot be edited).
10. Shut down the old machine.

# 4 Installing QlikView Server

**Note!** If Microsoft IIS is to be used as web server, it must be installed prior to QlikView Server.

**Note!** IPv4 is required for installation of QlikView Server. IPv6 is currently unsupported.

It is recommended not to move folder locations after the QlikView Server installation is complete, since many settings depend on the initial file locations. If the location of QlikView Server has to be changed after the installation, proceed as follows:

1. Run the QlikView Server installation executable:
   - Microsoft Windows x86 version: `QlikViewServer_x86Setup.exe`
   - Microsoft Windows x64 version: `QlikViewServer_x64Setup.exe`
   - Microsoft Windows Server 2012/Windows 8 (and later): `QlikViewServer_Win2012andUp.exe`
2. If the User Account Control dialog is displayed, click **Yes** to allow the program to make changes on this computer.
3. Click **Next** in the Welcome dialog.
4. Select the region for the location of the server. Click **Next** to continue.
5. Read the license agreement, select **I accept the terms in the license agreement**, and click **Next** to continue.
6. Enter the customer information for QlikView Server. Click **Next** to continue.
7. All files are installed in the specified folder. To change the root folder for the installed files, click **Change** to specify the preferred location. Finally, click **Next** to continue.
8. Select the type of installation you want to perform:
   - **Full installation, Single machine with QlikView Webserver:** Used to run all components on a single machine with QlikView Web Server as web server.
   - **Full installation, Single machine with Microsoft IIS:** Used to run all components on a single machine with Microsoft IIS as web server. This option is only available if IIS is installed on the target machine.
   - **Custom installation, select profiles:** If this option is selected you select the profiles you want to be included in the installation from the Profiles section in the dialog:
     - **QlikView Server:** Installs QlikView Server, Directory Service Connector, and the QlikView Server example documents.
     - **Reload/ Distribute Engine:** Installs the Reload Engine and the QlikView Distribution Service.
     - **Management Console:** Installs the QlikView Management Service together with the QlikView Management Console (QMC).
     - **Webserver:** Installs the QlikView Web Server.

   To make further configuration of features to be installed, click **Config**. When done, click **Next**.

   To use pre-defined configuration of features, click **Next**.
9. Set the account that the QlikView Server and Publisher services are to run under. Click **Next** to continue.

   **Note!** If using a local administrator account on Windows XP Professional x64 SP2 that is not part of a domain, the installation program cannot resolve the account. This means that the account for the services in **Computer Manager** has to be set manually.

   You can also select **I want to specify the account to be used for the services later**.
10. Select the IIS Website from the drop-down list and click **Next**.

    **Note!** This step is only applicable if **Full installation, Single machine with Microsoft IIS** was selected in **Step 8**. If not, proceed directly to the next step.
11. Select the Service Authentication method:
    - **Use digital certificates:** Authenticate communication between QlikView servers using digital certificates and SSL. This alternative is recommended in environments where not all servers have access to a common Windows Active Directory or when the security provided by certificate authentication is required. Note that digital certificates are **only** supported by Windows Server 2008 R2 and later.
    - **Use QlikView Administrators Group:** Authenticate communication between QlikView services based on membership in the local Windows group QlikViewAdministrators. This alternative can be used in

environments where all servers that are part of the QlikView installation can authenticate using a common Windows Active Directory.

Click **Next** to continue.

12. Click **Install** to start the installation.

**Note!** This may take several minutes to complete.

13. Click **Finish** when the installation is complete.
14. Log off from Windows® and then log on again, so that group memberships added during the installation are updated.

**Note!** It may be sufficient to log off from Windows and then log on again. However, it is recommended to restart the machine to enable the QlikView Server functionality.

# 4.1 Logging the Installation

The setup procedure is logged when running the QlikView Server installation executable. The log files are as follows:

- Microsoft Windows x86 version: `QlikViewServerx86.wil`
- Microsoft Windows x64 version: `QlikViewServerx64.wil`
- Microsoft Windows Server 2012/Windows 8 (and later): `QlikViewServer_Win2012andUp.wil`

The log files are stored in the `Temp` folder of the user (for example, `%UserProfile%\AppData\Local\Temp`). Each time an installation is executed, a new file is generated, over-writing the previous log file.

# 4.2 Obtaining the MSI package

If the MSI package is needed for the installation, proceed as follows to extract it from the `.exe` file:

1. Start the installation from the `.exe` file and wait until the first dialog opens.
2. Locate the MSI file (often stored with a random name, for example, `ed34g.msi`) in the `Temp` folder in `%UserProfile%\AppData\Local` (`C:\Documents and Settings\username\Local Settings` on pre-Windows Vista systems).
3. Copy the `.msi` file to another location.
4. Exit the `.exe` installation.
5. Install QlikView Server using the `.msi` file. See *Silent Installation (page 93)* for information on how to perform a silent installation. For additional information, see *Deploying MSI Packages with Group Policies (page 129)*.

# 4.3 Completing the Installation

After successfully installing QlikView Server, a license must be registered in QlikView Management Console (QMC) to activate the installed software.

**Note!** If access is denied when starting QMC, log off from Windows and then log on again, so that group memberships added during the installation are updated.

**Note!** Running real-time anti-virus protection on the server degrades the performance of QlikView Server. It is recommended that the user documents, source documents, log directories, and `.pgo` files are excluded from the anti-virus scanning.

## Running Microsoft IIS

### Handling Timeouts

**Note!** This is only needed when using very large QlikView documents that return timeouts.

Proceed as follows to handle timeouts:

1. Open the `%ProgramFiles%\QlikView\Server\QlikViewClients\QlikViewAjax\web.config` file in a text editor (for example, Notepad).
2. Search for the following text:
   `<httpRuntime requestValidationMode="2.0" />`
3. Edit the text so that it becomes:
   `<httpRuntime requestValidationMode="2.0" executionTimeout="900"/>`
4. Save the file.

## Enabling ASP.NET

If Microsoft IIS is used as web server in a Windows Server 2003 (or later) environment, enable ASP.NET to ensure proper operation of the QlikView Server sample pages and the extended functions (for example, QlikView Server tunnel).

## Optimizing the Performance

To optimize the performance when running Microsoft IIS and AJAX, turn on compression in the web server.

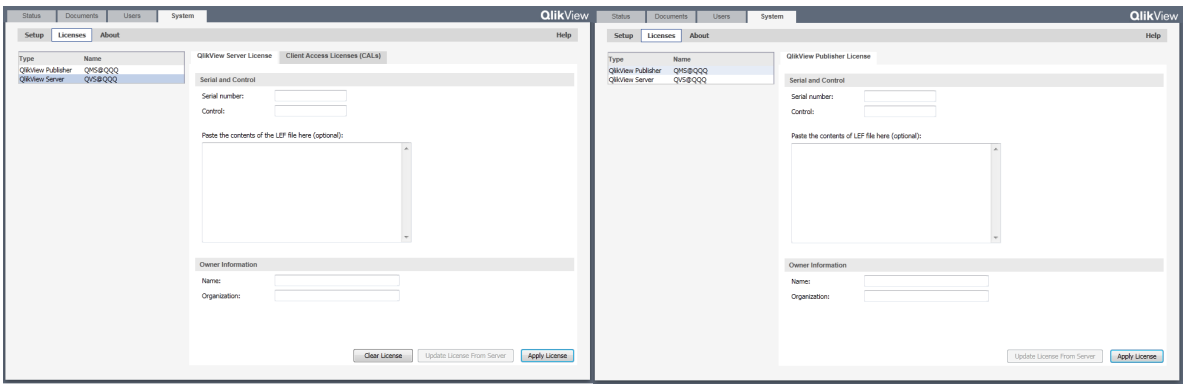
For information on how to configure IIS 6, see
**http://technet.microsoft.com/en-us/library/cc730629%28WS.10%29.aspx**.
For information on how to configure IIS 7, see
**http://technet.microsoft.com/en-us/library/cc782942%28WS.10%29.aspx**.

# Licensing

The licensing is used to authenticate QlikView Server and allow it to run on a specific machine.

Go to **System**>**Licenses** in QMC, select a QlikView Server or Publisher, and fill in the **Serial number** and **Control** fields on the **QlikView Server License** or **QlikView Publisher License** tab (depending on whether QlikView Server or Publisher was chosen).

**Note!** The license is checked every time a document is opened. If the time limit specified by the License Enabler File (LEF) is reached, the QVS automatically enters offline mode, which means that it is reachable from the QMC, but not operational.




*QlikView Server/Publisher License tab in QMC*

The License Enabler File (LEF), `lef.txt`, for QlikView Server is automatically saved in `%ProgramData%\QlikTech` (`C:\Documents and Settings\All Users\Application Data\QlikTech` on pre-Windows Vista systems).

The `PubLef.txt` file for QlikView Publisher is saved in `%ProgramData%\QlikTech\ManagementService\Publisher LEF` (`C:\Documents and Settings\All Users\Application Data\QlikTech\ManagementService\Publisher LEF` on pre-Windows Vista systems).

Click **Update License from Server** to download a new `lef.txt` file from the QlikView LEF server. This is primarily used when updating the number of Client Access Licenses (CALs).

If the LEF information cannot be accessed through the Internet, it can be obtained from the local vendor. In that case, copy the entire `lef.txt` file to the location mentioned above, or paste the LEF data using the corresponding field on the QlikView Server/Publisher License tab in QMC. Contact the local vendor for specific instructions.

# 5 Building a Farm

Server farms can be used to provide additional performance, redundancy, and security in place of a single server solution.

## 5.1 Planning

Before starting the actual installation, planning is needed. The following items have to be considered:

- Trust mechanism
- Web server (QlikView Web Server or Microsoft IIS)
- Redundancy level
- Account to run the services under
- QVPR format (XML or SQL)
- User directory
- User authentication
- Firewalls

### Trust Mechanism

Trust mechanisms are provided with Windows groups or certificates.

Windows groups can easily be deployed, if all services reside in a single Active Directory (AD). If encrypted communication is needed, it can be added manually.

Certificates provide for trust mechanisms in cross-domain environments and can also provide SSL encryption.

### Web Server

QlikView Web Server is intended for use when the web server is not needed for other purposes. It is lightweight and easy to manage, but at the same time limited to support the tasks needed by a QlikView installation.

A Microsoft IIS-hosted web server is recommended, if:

- More flexibility or more advanced tuning is required
- The web server is to be used for other tasks than QlikView
- An authorization scheme not available out-of-the-box is required

### Redundancy Level

The redundancy level is mainly a question of clustering and/or having multiple machines running the same service. All services except QlikView Management Service (QMS) can be installed on multiple machines. In addition, QlikView Server (QVS), QlikView Distribution Service (QDS), and Directory Service Connector (DSC) can be clustered.

### Account to Run the Services Under

A dedicated account should be created to manage the QlikView services. The account should be assigned during the installation, with proper privileges, see *Security Overview (page 69)*. It is recommended that the same account is used for all services.

### QVPR Format

The choice of QVPR format is based on reasons outside the QlikView product (for example, backup and availability). The installation always starts in XML mode.

### User Directory

QlikView defaults to Windows users (that is, NTFS mode). If non-Windows users are to be given access (other than anonymously), QlikView Server must run in Document Metadata Service (DMS) mode.

DMS mode may also be preferable for other reasons, see *Document Level Authorization (page 79)*.

### User Authentication

QlikView supports multiple authentication schemes. Additional schemes may require ASPX development and the possible use of Microsoft IIS for web services.

For information on the available authentication schemes, see *Authentication (page 73)*.

### Firewalls

Make sure that the services are able to communicate (for example, by opening the appropriate ports in the firewalls). For information on the ports, see *Service by Service (page 53)*.

## 5.2 Root/First Install

Before starting, make sure that the appropriate service account (or accounts) is set up and available on the machines where the services are to be installed.

In all installations, there must exist exactly one QMS, which must be installed first. Note that the QMS must be able to communicate with all the subsequently installed services.

If more services are to run on the same server, they can be installed at the same time.

## 5.3 Adding Services on Other Machines

The next step is to install the other services on the other servers. If more services are to run on the same server, they can be installed at the same time. The order in which the services are added is not important.

When the services have been installed, it is time to return to QlikView Management Console (QMC) and configure the services. This is done on the System tab. The first step is to add the services. Make sure to note the differences between building out a cluster and creating a brand new cluster.

## 5.4 Clustering

This section provides an overview of how create a QlikView Server cluster. For additional information, see *Clustering QlikView Servers (page 101)* and *Clustering QlikView Publisher (page 109)*.

**Note!** Do *not* mix architectures – that is, 32-bit (x86) and 64-bit (x64) – within a cluster.

### QlikView Server

For the QlikView Server cluster to work properly, it is important to set **System**>**Setup**>**QVS resource**>**Folders**>**Root Folder** to a common shared folder. In addition, **Alternate Temporary Files Folder Path** must be set to a common shared folder (separate from the root folder).

If extensions are used, it simplifies management if **Alternate Extension Path** is set to a common shared folder.

It is also common practice to set **System**>**Setup**>**QVS resource**>**Logging**>**Log Folder** to a common place, but this is not strictly necessary.

**Note!** The root folder must *not* be used for anything else than cluster files (that is, `.pgo` files) and user documents.

# QlikView Distribution Service

For a cluster of QDSs, **System**>**Setup**>**General**>**Application Data Folder** must be set to a common shared folder. In addition, **Source Folders** must be common shared folders.

# Directory Service Connector

A cluster of DSCs does not need any specific settings. The difference between clustered and non-clustered DSCs is whether the settings are shared or not.

# QlikView Web Server

Multiple web servers can be set up, but they are always configured independently (that is, they are never clustered). Note that it is uncommon, but from a technical perspective possible, to have some web servers running QlikView Web Server (QVWS) and some Microsoft IIS.

## Tunneling Using Microsoft IIS

Tunneling is used by Windows native clients (QlikView Desktop, the OEM OCX, and the Internet Explorer plugin) and needed when the clients cannot communicate with QlikView Server on port 4747 (most likely due to a firewall blocking the traffic):

- QVWS: No extra settings are required.
- Microsoft IIS: The `QVSTunnel.dll` file must be added as an ISAPI filter.

Proceed as follows to set up tunneling for Microsoft IIS 7:

1. Open the Internet Information Services Manager.
2. Select the IIS top node.
3. Open the ISAPI and CGI Restrictions dialog.
4. Select **Add** in the Actions pane and browse to the location of `QVSTunnel.dll`.
5. Provide a description of the instance and check the **Allow extension path to execute** box.
6. Open the site that is to host the QlikView Server and Publisher pages and click **Scripts**.
7. Open the Handler Mappings dialog.
8. Locate ISAPI dll and select **Edit Features Permission** in the Actions pane.
9. Click **Execute** in the dialog that opens.

The following entries are required in the registry when the QVS and Microsoft IIS are located on different machines:

`[HKEY_LOCAL_MACHINE\SOFTWARE\QlikTech\QlikTunnel]`

- “QVSPort”=dword:000012a6
- “QVSServer”=“QvsHost”

**Note!** If the entries do not already exist in the registry, they have to be added manually.

Test the QlikView Server tunnel by entering the following URL in a client browser window:

`http://<Servername>/scripts/qvstunnel.dll?test`

`Servername` is the web server. If the tunnel is correctly set up, the web page returns a message (that tunneling is available) and the QlikView Server version number.

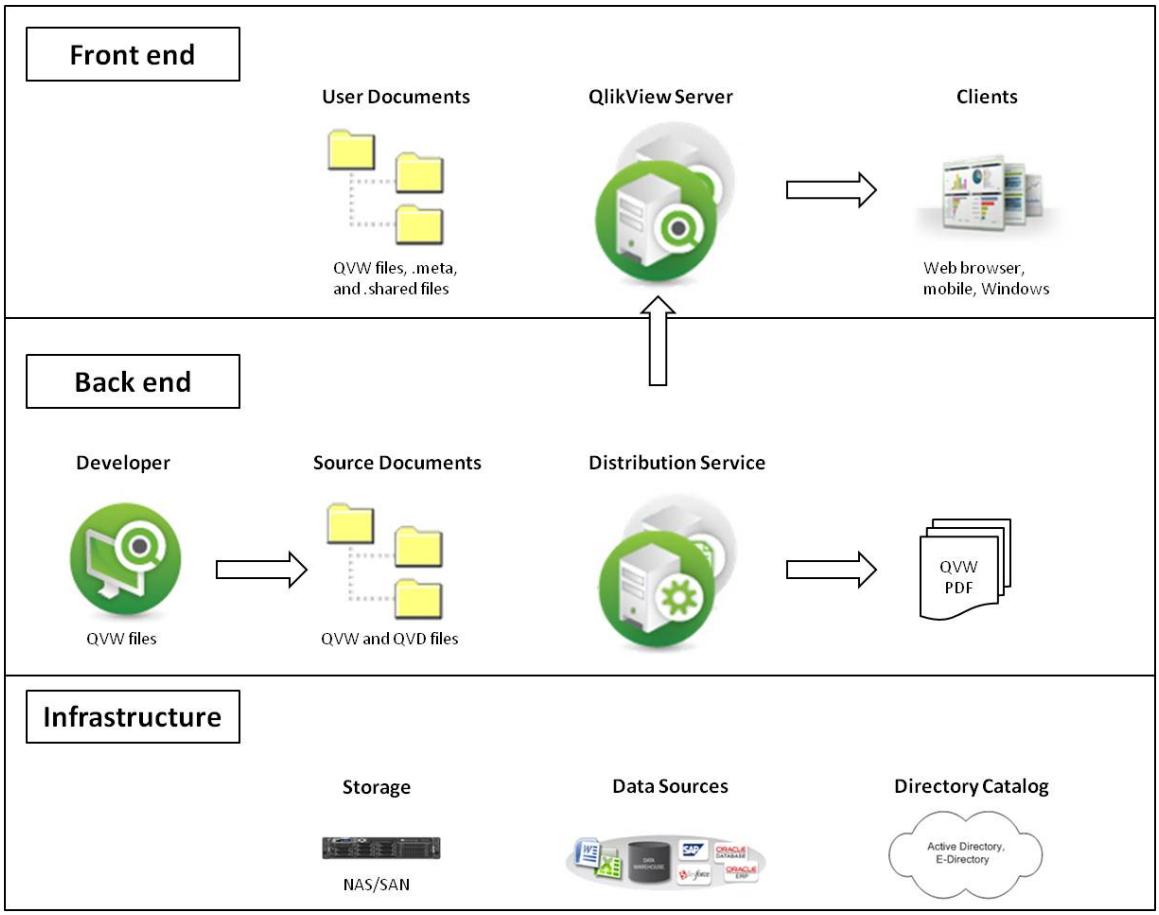
# Part 3 Architecture

# 6 Roles

The overall architecture of a QlikView installation reflects the separation of roles.

## 6.1 QlikView with Publisher

The figure below shows a QlikView deployment with Publisher containing the location of the QlikView components.




*QlikView deployment with Publisher containing the location of the QlikView components*

## Front End

The front end is where end users interact with the documents and data that they are authorized to see via QlikView Server. The front end contains the QlikView user documents that typically have been created via QlikView Publisher at the back end. All communication between the client and server takes place here and QlikView Server is fully responsible for the client authorization.

The front end relies on infrastructure resources (for example, Windows-based File Share for clustering).

**Note!** QlikView Server currently only conforms with Windows File Share or a Windows-based NAS. This means that storage must be owned, governed, and shared by a Windows operating system instance (typically accessed using a path like `\\<servername>\<share>`).

Authentication of end users is (with exception of the built-in Custom Users) handled outside QlikView.

## Back End

The back end is where the QlikView source documents, created using QlikView Developer, reside. These source files contain scripts to extract data from various data sources (for example, data warehouses, Microsoft Excel® files, SAP®, and Salesforce.com®). This extraction sometimes involves intermediate files (QVD files). The main QlikView component that performs the loading and distribution at the back end is the Distribution Service. Within the back end, the Windows file system is always in charge of authorization (that is, QlikView is not responsible for any access privileges).

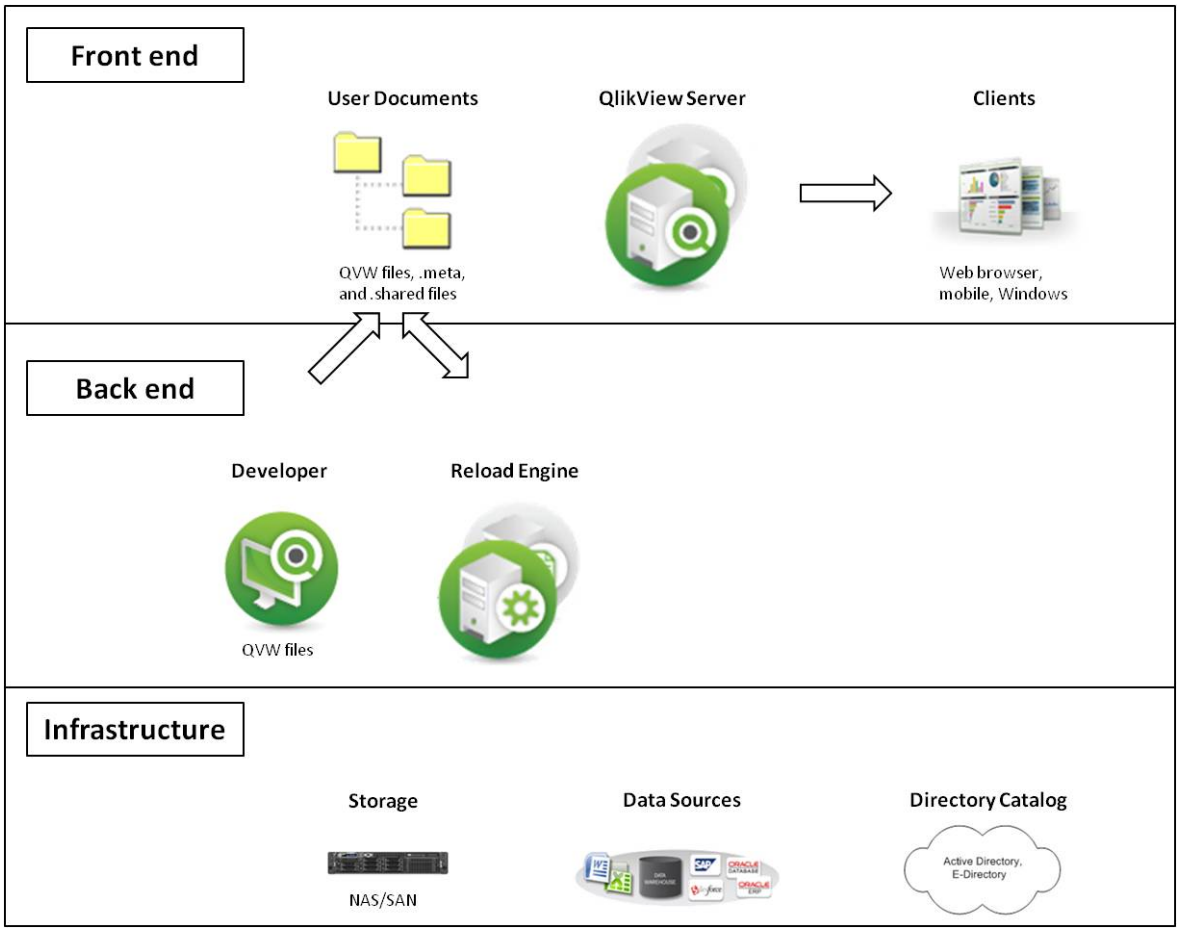
The back end uses the infrastructure resources for clustering (for example, Windows-based File Share) and may also use resources like SMTP servers and directory catalogs.

**Note!** QlikView Server currently only conforms with Windows File Share or a Windows-based NAS. This means that storage must be owned, governed, and shared by a Windows operating system instance (typically accessed using a path like `\\<servername>\<share>`).

As depicted here, both the back end and front end are suitable for development, testing, and deployment.

# 6.2 QlikView without Publisher

Without Publisher, the QlikView architecture becomes more restrictive. All distribution and reduction facilities are removed and replaced by a reload directly on the user documents. Without the distribution abilities of Publisher, developers need to manually deploy the `.qvw` file behind the server.



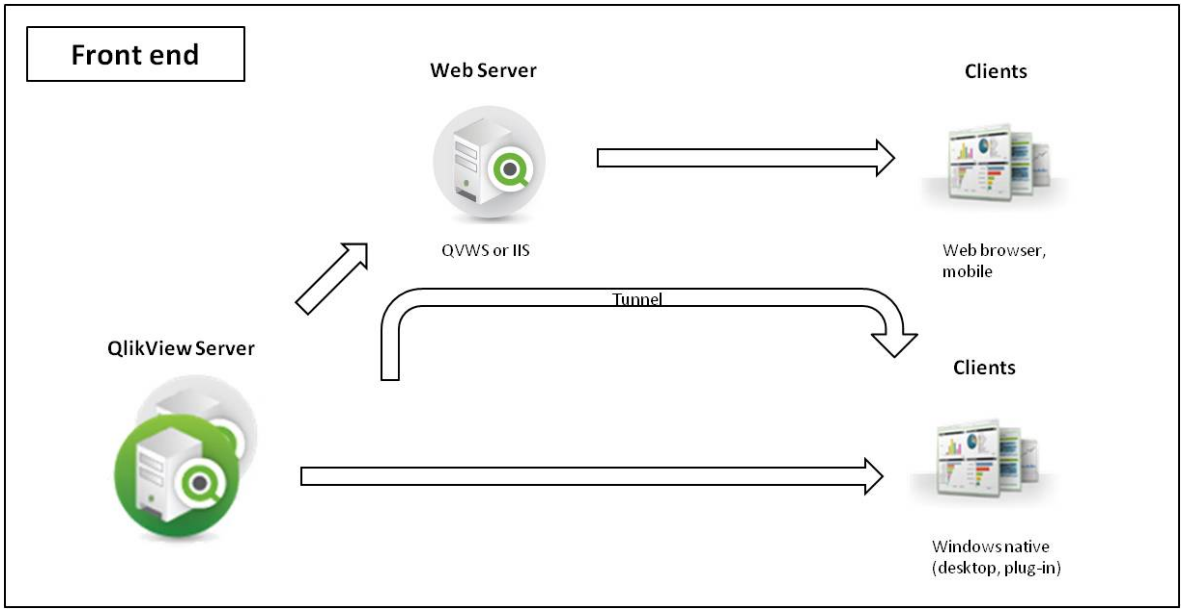

*QlikView architecture without Publisher*

## 6.3 QlikView Server

The number of servers (clustered or not) within an installation is only limited by the license. It is, however, not feasible to run more than one QVS process per server (physical or virtual). QVS is designed to make the most of the resources available to it. Notably the QVS keeps as many calculation results as possible cached in memory to keep the response times to a minimum.

### QlikView Server – Client Communication

The QlikView Server – client communication architecture requires three primary processes, which must be able to communicate with each other in a consistent and secure manner. This interaction can potentially involve multiple machines and multiple network connections, as well as other subordinate processes.




*QlikView Server – client communication*

The three primary processes are described below.

| Process | Description |
|---|---|
| QVS | Provides QlikView functionality to the client. The machine that hosts this service must be running a Microsoft Windows operating system. |
| Client | Runs in a web browser or an application shell that provides a container for the client code. The client communicates with QVS either directly or through the web server to provide the QlikView interface and functionality to the end user. |
| Web server | Runs an http server, which can be used to serve html web pages to the client, assists with authentication of the user, and enables communication between the client and QVS. |

With the exception of Custom Users, the authentication of client users is done outside QlikView using, for example, Windows authentication. See *QlikView Server Authentication Using Custom Users (page 76)* for information on authentication of Custom Users.

The protocols defined for client communication with QVS are listed below.

| Protocol | Description |
|---|---|
| QlikView Protocol (QVP) | Encrypted, binary, and TCP-based; communicates directly with QVS on port 4747. |
| QVPX | XML-based; communicates with the QVS using http/https through a web server. |

Windows clients (`.exe`/`.ocx`) communicate directly with QVS using QVP on port 4747. These clients do not require a web server to establish and maintain a connection with QVS.

The AJAX client and mobile clients do not communicate directly with QVS. They establish and maintain a connection using the QVPX protocol through a web server, QlikView Web Server (QVWS) or Microsoft IIS. This is normally done using port 80 (http). The web server, in turn, communicates with QVS using the QVPX2 protocol on port 4747.

The default installation settings for QVS use QVWS, not IIS. QVWS shares port 80 with IIS on Windows Vista (and later) and Windows Server 2003 (and later). On Windows XP, only one of the two web servers can use port 80. If both are configured to run, they must be assigned different ports.

## QlikView Server – User Document

For a user to open a document, it is required that:

- There is a Client Access License (CAL) for the user
- The user has access to the document

The user documents are always read by QVS and thus technically only need to be readable by the account running QVS. The access rights are either stored in the ACL list of the document (when QVS runs in NTFS mode) or in the `.META` file (when QVS runs in Document Metadata Service – that is, DMS – mode). These settings are part of the distribution from the back end.

Items (for example, layout, reports, bookmarks, annotations, and input field values) created by end users are stored in `.Shared` files. `.Shared` files are *not* replaced by the distribution from the back end.

# 6.4 Web Server

QlikView Web Server (QVWS) is included as part of the QlikView Server installation. The web server can act as a standalone service to fulfill the need of many QlikView Server installations.

As an alternative, a Microsoft IIS solution that provides more flexibility, additional authentication schemes, and web services for applications other than QlikView Server can be deployed. When IIS is used, a special service, QlikView Settings Service, that handles management calls is installed.

Other web servers can be used in a QVS environment, but at some point the traffic targeting QVS has to go through either QVWS or the dedicated ASPX pages on IIS.

The QlikView Web Server component (either QVWS or IIS-based) performs several tasks:

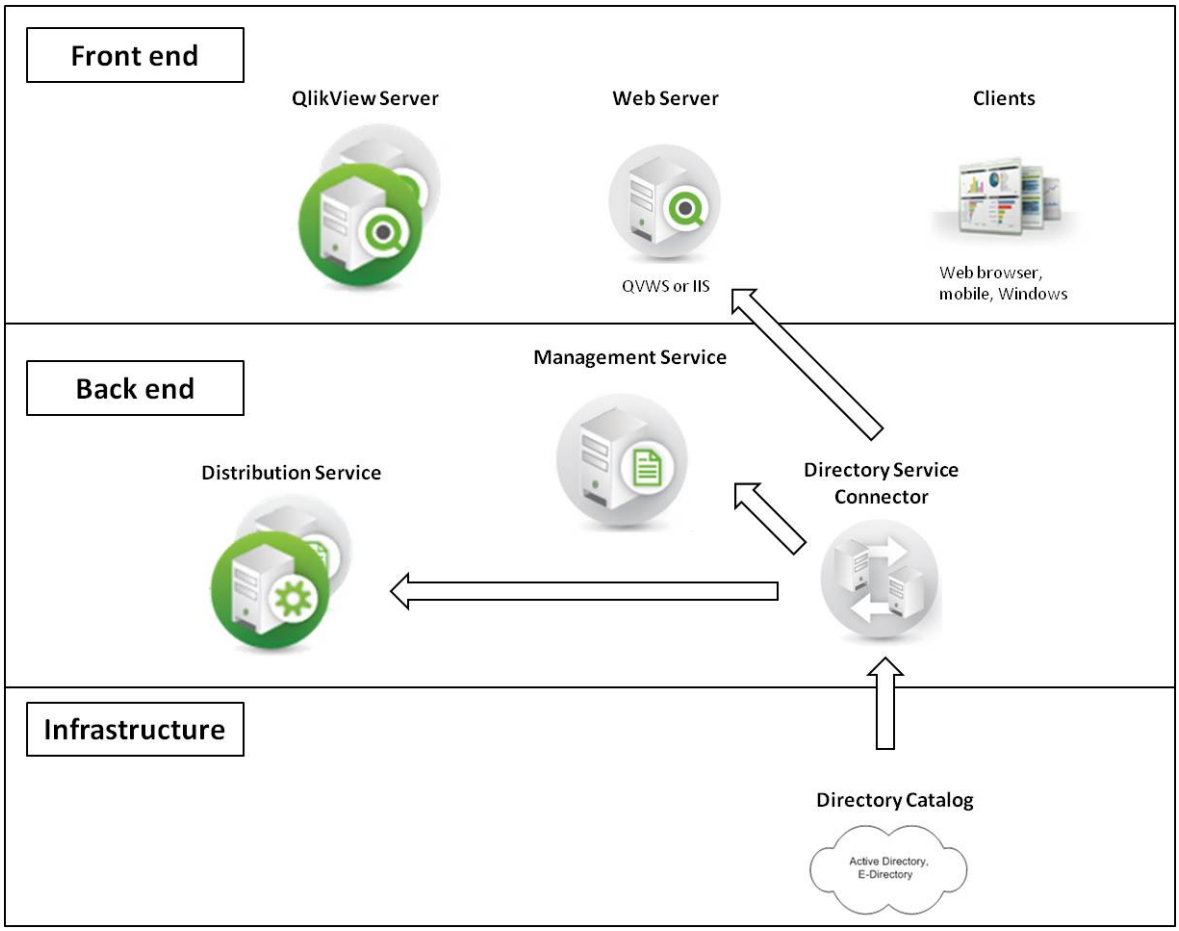
- Handles the AccessPoint back end
- Transforms/routes traffic between stateless http and to/from the session-based communication with QVS
- Handles load balancing of QVS clusters
- Serves static content (optional)
- Handles authorization of Windows-authenticated users
- Handles authentication of Custom Users (optional)
- Handles group resolution through Windows or Directory Service Connector (DSC) (optional)

## QlikView Server Tunnel

If the QVS communication port (4747) is blocked in the network firewall, Windows clients attempt to re-route their connection through port 80 (http). This connection path must then include the QVWS, or be installed on Microsoft IIS, so that QVS tunnel communication can be established.

# 6.5 Directory Service Connector

The figure below shows the information flow. The Directory Service Connector (DSC) is responsible for retrieving user information related to end users from a variety of sources, including (but not limited to) Active Directory, LDAP, ODBC, and Custom Users.

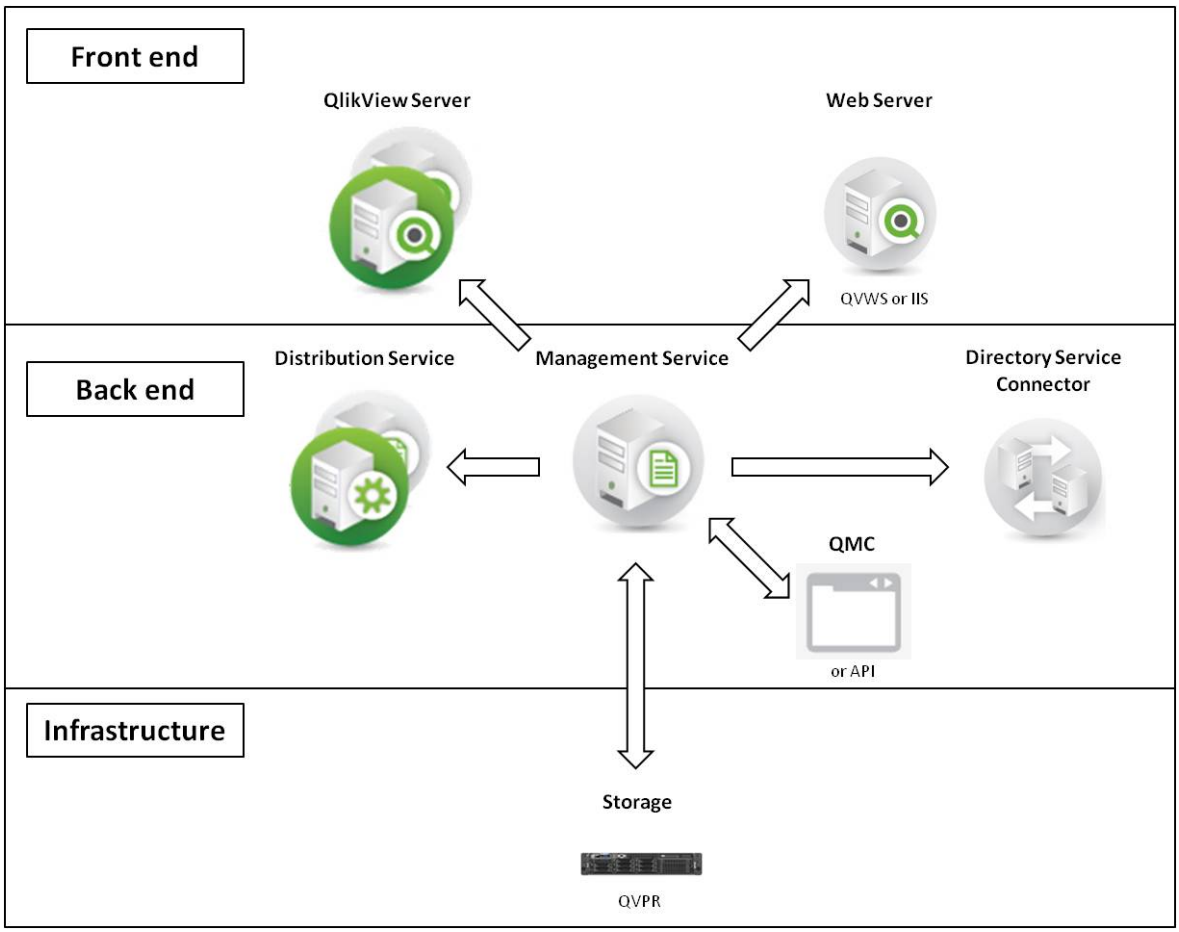



*Directory Service Connector*

The web server uses DSC for group resolution, the Distribution Service uses it to look up e-mail addresses or UIDs during distribution, and the Management Service uses it to help the administrator find users and groups.

# 6.6 Management Service

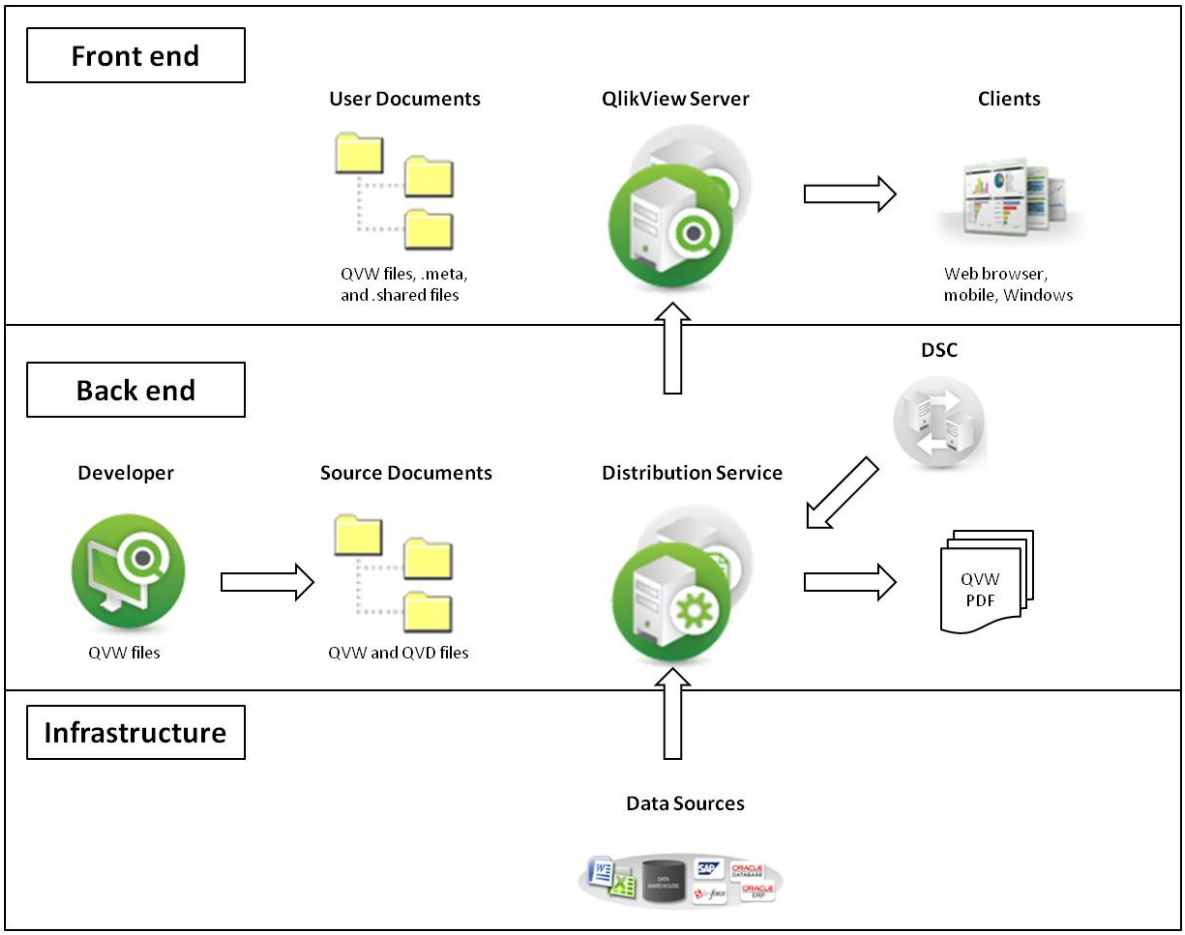
The Management Service is the entry point for all management, both through QlikView Management Console and the QlikView APIs.




*Management Service*

The QlikView Management Service (QMS) keeps settings in a database of its own, the QVPR. The QVPR is by default stored as XML files – an alternative is storing the settings in an SQL database. An installation can only have a single instance of QMS active. Active/passive failover should be used for redundancy. Note that no other service needs QMS to be running.

# 6.7 Distribution Service




*Distribution Service*

The Distribution Service works with the source documents to produce:

- User documents
- `.qvw` files for distribution to a folder or via e-mail
- `.pdf` documents for distribution to a folder or via e-mail

The chain of events up to the final distribution involves one or many of the following tasks:

1. Data is loaded from one or more data sources (including QVD) into one or more `.qvw` or `.qvd` files.
2. A document is reduced into one or more smaller documents.
3. Attributes and usage rules are added (applicable only when distributed to a QVS).

The Distribution Service performs the tasks according to defined schedules and/or as responses to events.

# 6.8 Reload Engine

In the absence of a Publisher license, the Reload Engine provides a subset of the Publisher distribution services. The Reload Engine only reloads user documents and the settings are defined directly in the user documents.

# 7 Logging

All alerts from the QlikView services appear in the Windows event log.

## 7.1 Logging from QlikView Server

Detailed session logs are found in the logging directory, which is specified on the **System**>**Setup**>**Logging** tab in QlikView Management Console (QMC). The default location is `%ProgramData%\QlikTech\QlikViewServer` (`C:\Documents and Settings\All Users\Application Data\QlikTech\QlikViewServer` on pre-Windows Vista systems).

Log files can be set to split (that is, create new) daily, weekly, monthly, yearly, or never. Performance log intervals can be set from one minute and higher.

**Note!** Setting the interval to be very small, for example, only one minute, may negatively impact the performance.

## 7.2 Session Log

A session is defined as a single user connected to a single document.

**Note!** The session log is updated each time a session *ends*. This means no log entry is created when a session starts.

The file name of the session log is `Sessions*.log`, where `*` reflects the server name and the split interval. Each entry of the session log contains the fields listed below.

| Field | Description |
|---|---|
| Exe Type | Type of QVS build.<br>Example: “RLS32” = 32-bit release build |
| Exe Version | Full version number of QVS.<br>Example: “11.00.11076.0409.10” |
| Server Started | Date and time when QVS was started. |
| Timestamp | Date and time when the log entry was created. |
| Document | QlikView document that was accessed. |
| Document Timestamp | File timestamp of the document that was accessed. |
| QlikView User | QlikView section access user ID (if used). |
| Exit Reason | Reason for session termination:<br>• “Socket closed” = Client-induced termination<br>• “LRU” = Terminated as Least Recently Used in favor of new user<br>• “Shutdown” = Server-induced termination for other reasons<br>**Note!** This is not a complete list, as the exit value in some cases comes from the operating system. |
| Session Start | Time when the session was started. |

| Field | Description |
|---|---|
| Session Duration | Duration of session in hours:minutes:seconds. |
| CPU Spent (s) | CPU seconds spent by the session. |
| Bytes Received | Bytes received by the server during the session. |
| Bytes Sent | Bytes sent by the server during the session. |
| Calls | Number of QlikView calls during the session (bidirectional). |
| Selections | Number of QlikView selections made during the session. |
| Authenticated User | Authenticated Windows NT® user ID (if any). |
| Identifying User | Client user identification. |
| Client Machine Identification | Client machine identification. |
| Serial Number | Serial number of the QlikView client (installed clients only, that is, QlikView Desktop and Internet Explorer plugin). |
| Client Type | Client type used:<br>• "Windows Exe" = QlikView Desktop and Internet Explorer plugin<br>• "Ajax" = all clients that use the QVPX protocol<br>• "Unknown" |
| Client Build Version | Build version of the QlikView client. |
| Secure Protocol | Secure protocol used:<br>• "On" when encrypted communication is used (typically Windows clients).<br>• "Off" when non-encrypted communication is used. |
| Tunnel Protocol | "Tunnel" when QVS tunnel communication is used. |
| Server Port | Port used by the server. |
| Client Address | Client IP number. |
| Client Port | Client port. |
| CAL Type | Client Access License (CAL) type:<br>• "User" = Named User CAL<br>• "Session" = Session CAL<br>• "Usage" = Usage CAL<br>• "Document" = Document CAL |
| CAL Usage Count | Number of Usage CALs. |

# 7.3 Performance Log

The performance log is updated at the interval specified on the **System**>**Setup**>**Logging** tab in QMC. The default interval is five minutes. Additional entries are added whenever the server is started or stopped. The file name of the session log is `Performance*.log`, where `*` reflects the server name and the split interval. Each entry of the log contains the fields listed below.

| Field | Description |
|---|---|
| Exe Type | Type of QVS build.<br>Example: “RLS32” = 32-bit release build |
| Exe Version | Full version number of QVS.<br>Example: “11.00.11076.0409.10” |
| Server Started | Date and time when QVS was started. |
| Timestamp | Date and time when the log entry was created. |
| EntryType | Entry type:<br>• “Server starting” = Startup<br>• “Normal” = Normal interval log entry<br>• “Server shutting down” = Shutdown |
| ActiveDocSessions | Number of document sessions* that has shown activity during the interval and still exists at the end of the interval. |
| DocSessions | Total number of document sessions* that exists at the end of the interval. |
| ActiveAnonymousDocSessions | Number of document sessions* with anonymous user that has shown activity during the interval and still exists at the end of the interval. |
| AnonymousDocSessions | Total number of document sessions* with anonymous user that exists at the end of the interval. |
| ActiveTunneledDocSessions | Number of document sessions* with tunneled connection that has shown activity during the interval and still exists at the end of the interval. |
| TunneledDocSessions | Total number of document sessions* with tunneled connection that exists at the end of the interval. |
| DocSessionStarts | Number of document sessions* that has been initiated during the interval. |
| ActiveDocs | Number of documents loaded at the end of the interval in which there has been user activity during the interval. |
| RefDocs | Number of documents loaded at the end of the interval for which there is a session at the end of the interval. |
| LoadedDocs | Total number of documents loaded at the end of the interval. |
| DocLoads | Number of new documents loaded during the interval. |

| Field | Description |
|---|---|
| DocLoadFails | Number of documents that has failed to load during the interval. |
| Calls | Total number of calls to QVS during the interval. |
| Selections | Number of selection calls during the interval. |
| ActiveIpAddrs | Number of distinct IP addresses that has been active during the interval and still exists at the end of the interval.<br>**Note!** Tunneled sessions and multiple users originating from the same IP cannot be distinguished. |
| IpAddrs | Total number of distinct IP addresses connected at the end of the interval.<br>**Note!** Tunneled sessions and multiple users originating from the same IP cannot be distinguished. |
| ActiveUsers | Number of distinct NT users that has been active during the interval and still exists at the end of the interval.<br>**Note!** Anonymous users cannot be distinguished. |
| Users | Total number of distinct NT users connected at the end of the interval.<br>**Note!** Anonymous users cannot be distinguished. |
| CPULoad | Average CPU load from QVS during the interval. |
| VMAllocated(MB) | Size in MB of the virtual memory allocated by QVS at the end of the interval**. |
| VMCommitted(MB) | Size in MB of the virtual memory actually used by QVS at the end of the interval. This number is part of VMAllocated(MB) and should not exceed the size of the physical memory in order to avoid unacceptable response times. |
| VMFree(MB) | Size in MB of the unallocated virtual memory available to QVS**. |
| VMLargestFreeBlock(MB) | Size in MB of the largest contiguous block of unallocated virtual memory available to QVS. This number is part of VMFree(MB). |
| UsageCalBalance | “-1.00” = There are no Usage CALs. |

*One user + one document = One document session.

**VMAllocated(MB) + VMFree(MB) = Total maximum virtual memory space available to the QVS process.

## 7.4 Event Log

The event log is updated each time a log entry is made in the Windows event log by QVS. The stored information is a mirror of the information written to the Windows event log. The file name of the event log is `Events*.log`, where `*` reflects the server name and the split interval.

Use the **Event Log Verbosity** radio buttons on the **System**>**Setup**>**QlikView Servers**>**Logging** tab in the QMC to set the verbosity level. Depending on the verbosity level selected, the following entries are written to the Event log:

- **Low**: Error messages
- **Medium**: Error and warning messages
- **High**: Error, warning, and information messages

Each entry of the log contains the fields listed below.

| Field | Description |
|---|---|
| Server Started | Date and time when QVS was started. |
| Timestamp | Date and time when the log entry was created. |
| SeverityID | ID for the severity level:<br>1 = Error<br>2 = Warning<br>4 = Information |
| EventID | Unique ID for the event type. |
| Severity | Event severity level:<br>• Error<br>• Information<br>• Warning |
| Message | Event description. |

# 7.5 End-user Audit Log

The end-user audit log contains information on user selections, including cleared selections, activated sheets, application of bookmarks, accessed reports, and maximized objects.

A log file called `AUDIT_<machinename>` is saved to `%ProgramData%\QlikTech\QlikViewServer` (`C:\Documents and Settings\All Users\Application Data\QlikTech\QlikViewServer` on pre-Windows Vista systems).

**Note!** Tick the **Enable Extensive Audit Logging** check box on the **System**>**Setup**>**QlikView Servers**>**Logging** tab in the QMC to enable detailed audit logging (for example, logging of all selections that come with a bookmark). However, the logging of user selections in QVS is based on how the current selections object works and therefore larger selections may not be logged in detail.

| Field | Description |
|---|---|
| Server started | Date and time when QVS was started. |
| Timestamp | Date and time when the log entry was created. |
| Document | Path and name of the document that was accessed. |
| Type | Type of selection made (for example, “Selection” or “Bookmark”).<br>For an overview of the types available, see the table below. |
| User | User name. |

| Field | Description |
|---|---|
| Message | Information on the type of selection or application of bookmark that was made in the document (for example, “Apply Server\Bookmark15”).<br>For an overview of the messages that can be posted in this field, see the table below. |

The types and messages that can be posted in the Type and Message fields in the end-user audit log are listed below.

**Note!** In the end-user audit log, “XXX” and “YYY” are replaced with values from the QlikView document.

| Type | Message | Description |
|---|---|---|
| Bookmark | Apply XXX | Bookmark XXX was applied. |
| Bookmark Selection | XXX | Selection XXX was made because a bookmark was selected. Entries of this type are only logged when detailed audit logging is selected. |
| Export | Sheet Object XXX | Sheet object XXX was exported. |
| Maximize | Sheet Object XXX | Sheet object XXX was maximized. |
| Print | Sheet Object XXX | Sheet object XXX was printed. |
| Report | Accessed report XXX | Report XXX was accessed. |
| Selection | Clear All | All selections were cleared. |
| Selection | XXX | Selection XXX was made. |
| SendToExcel | Sheet Object XXX | Sheet object XXX was sent to Microsoft Excel. |
| Sheet | Activated sheet XXX | Sheet XXX was activated. |
| Session Collaboration | Session Collaboration Initiated, ID:XXX | A session collaboration with ID XXX was initiated. |
| Session Collaboration | Session Collaboration user XXX joined session, ID:YYY | User XXX joined the session collaboration with ID YYY. |
| Session Collaboration | Session Collaboration user XXX left session, ID:YYY | User XXX left the session collaboration with ID YYY. |

The following example shows the resulting log entry when a bookmark (“Bookmark01”) is selected. The log has been put in a table for better overview.

| Field | Value |
|---|---|
| Server started | 2013-05-06 10:17:33 |
| Timestamp | 2013-05-06 10:23:28 |

| | |
|---|---|
| Document | C:\ProgramData\QlikTech\Documents\Test.qvw |
| Type | Bookmark |
| User | QlikTech\jsmith |
| Message | Apply Server\Bookmark01 |

If detailed audit logging is selected, the log entry above may be followed by one or more log entries that detail the selections that were made because the bookmark was selected. In these log entries, the Type field is set to “Bookmark Selection”.

# 7.6 Manager Audit Log

The audit logging provides the possibility to track changes to tasks and settings in the system in order to see who made the changes and when they were made.

The audit logs are stored in `%ProgramData%\QlikTech\ManagementService\AuditLog`. One folder per table is created. Each folder contains one file per day with the changes made to the tasks. The logs are tab separated files.

The entries found in the logs are listed below.

| Entry | Description |
|---|---|
| `TransactionID` | Transaction ID, which is useful for keeping track of changes made simultaneously. |
| `ChangeType` | Type of operation, `Update` (new or changed entries) or `Delete` (entries have been deleted). |
| `ModifiedTime` | Time and date (in UTC) when the changes were made. |
| `ModifiedByUser` | The user that made the changes in the user interface. `System` means that the change was initiated by the system and not by any user. |
| `ID` | ID of the row (that was updated or deleted) in the table that was changed. |

The following example comes from the `AlertEmail` table. The log has been put in a table for better overview.

| | |
|---|---|
| `TransactionID` | 455a241d-8428-4dc7-ba67-4ae7cb21cf3d |
| `ChangeType` | Update |
| `ModifiedTime` | 2010-02-02 15:12:54 |
| `ModifiedByUser` | MyDomain\mjn |
| `ID` | b3745325-cee7-4fe7-b681-9c9efe22fc5c |
| `DistributionServiceID` | 8846d7dd-bb3f-4289-9c9b-b0ca71b7c3b2 |
| `EmailAddress` | mjn |

The following example comes from the `QDSCluster` table. Note that `TransactionID` is the same for both examples. This means that the changes were made simultaneously.

| | |
|---|---|
| TransactionID | 455a241d-8428-4dc7-ba67-4ae7cb21cf3d |
| ChangeType | Update |
| ModifiedTime | 2010-02-02 15:12:54 |
| ModifiedByUser | MyDomain\mjn |
| ID | a37f242c-6d80-42da-a10c-1742d2ec927f |
| DistributionServiceID | 8846d7dd-bb3f-4289-9c9b-b0ca71b7c3b2 |
| QDSWebAdress | http://computer-mjn:4720/qtxs.asmx |
| CurrentWorkorderID | 96bff2dc-f1ea-84d2-b6c4-ea58bf5c98e5 |

# 8 Documents, Data, and Tasks

## 8.1 User Documents

A user document is the document that an end user sees when accessing a document on QlikView Server (QVS). To fully identify a user document, both the QVS server/cluster and the path relative to the server have to be known. Technically, a user document consists of three files:

1. `.qvw` file that contains the data and layout.
2. `.META` file that contains:
    a. AccessPoint attributes
    b. Pre-load options
    c. Authorization (Document Metadata Service – that is, DMS – mode only)
3. `.Shared` file (see below)

**Note!** If the user document is distributed by the QlikView Distribution Service, both the `.qvw` and the data in the `.META` file are overwritten.

The access to user documents is controlled by QlikView Server.

### Shared Files

There are multiple objects available for user collaboration and sharing through QlikView Server:

- Bookmarks
- Sheet objects, including charts
- Reports
- Annotations

Each of these objects may be defined as a user object, available to authenticated users, regardless of access method or location, or a shared object, available to all users of the document through QVS.

The objects are configured and managed using QlikView Management Console (QMC).

Once QVS is enabled for server objects, any of the QVS object settings are checked, and the document is opened in QVS, a special database file is created and maintained in the same location as the QlikView document. The file has the same name as the QlikView document, but a `.Shared` file extension.

Example:

- QlikView document: `Presidents.qvw`
- QVS share file: `Presidents.qvw.Shared`

If the name of the QlikView document is changed, the `.Shared` file has to be manually renamed to match before opening the renamed QlikView document in QVS. This preserves the shared objects attached to the document.

When updating a Server object, report, bookmark, or input field data, the file is exclusively locked. Making a selection or simply activating the object does not lock the file and any number of servers can read the file at the same time. A partial lock is implemented so that different sections of the file may be updated simultaneously by different servers in a cluster.

The file is read once when the server opens the document, but it is not read again unless there are changes. All sessions share the same internal copy of the `.Shared` file (that is, opening a session generally does not require the file to be read from disk).

The server objects can be managed (for example, change of ownership or delete) on the **Documents**>**User Documents**>**Server**>**Server Objects** tab in QMC.

## 8.2 Source Data

Source data is any external data used to populate the data within a `.qvw` file. The source data is loaded to the `.qvw` at reload time, which can be done:

1. Through the QlikView Distribution Service
2. Through the Reload Engine
3. Manually by the developer

Access to source data is not required for end users to use the `.qvw` document through QVS once the `.qvw` file is populated.

## 8.3 Source Documents

Source documents are only applicable when a Publisher license is applied. Most source documents originate from a developer, others are created by the QlikView Distribution Service as part of the distribution process. QlikView Data files (QVD) can also be created as part of the distribution process as an intermediate step. A QVD file is a table of data stored in format that is optimized for speed when read by QlikView.

The access to source documents is governed by NTFS.

## 8.4 Tasks

Tasks can be used to perform a wide variety of operations and be chained together in any arbitrary pattern. The starting point when describing tasks is the transformation of a source document into a user document.

### Transforming Source Document into User Document

The transformation starts with a source document and ends in one or many user documents.

#### Source

A task is always tied to a source document, so the source is given.

#### Layout

The source document contains the layout, which is copied unchanged all the way to the user documents. The server side layout is associated with the user document and is also unchanged.

#### Reload

The data can be:

- Used as stored in the document (that is, no reload)
- Partly reloaded from the source (that is, require script preparation)
- Fully reloaded from the source, discarding any old data
- Reloaded in parts by use of “Script Parameters” (which require script preparation)

#### Reduce

The document can be reduced after reload. The reduction can either reduce the input into a smaller document (simple reduce) or split it up into several smaller documents (loop and reduce).

The reduction is based on a selection, either done directly in QMC or using bookmarks.

#### Distribution

Distribution requires a QlikView Publisher license.

The destination is defined as:

- A list of users and a folder on a QlikView Server
- A list of users and a folder in the file system
- A list of users (assuming their e-mail addresses are known)

**Note!** "Loop and distribute" must be used, if different content is to be distributed to different users. If not, the same document (or documents) is distributed to all.

## Information

Information can be associated with the document as part of the distribution to a server. The information is not moved with the document, if it is distributed to another location. The information is used in QlikView AccessPoint.

The following information can be associated with the document:

- Description
- Category
- Arbitrary name value pairs

## Server Settings

The settings for the document are distributed to a server. The settings are not moved with the document, if it is distributed to another location. The settings are enforced by QlikView Server.

Authorization enforced by the server (equal to all servers):

- The users authorized to create server objects
- The users authorized to download the document
- The users authorized to print and export the document to Microsoft Excel

Preferences applied by QlikView AccessPoint (equal to all servers):

- Internet Explorer plugin is recommended
- Mobile client is recommended
- AJAX client is recommended

Performance enforced by the server (equal to all servers):

- Audit logging
- Maximum open sessions
- Document timeout
- Session timeout

Availability (per server):

- Never
- On-demand
- Pre-loaded

# 9 Service by Service

This chapter describes the QlikView Server/Publisher components in detail.

## 9.1 QlikView Server

### Overview

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Server\QVS.exe` |
| **Data** | `%ProgramData%\QlikTech\QlikViewServer` |
| **Listens to** | QVP: 4747; QVP (tunneling): 4774; Broadcast: 14747; SNMP: 161 |
| **Uses/Controls** | - |
| **Used by** | QDS, QMS, QVWS, QlikView Desktop/Internet Explorer plugin/OCX |

### Files

#### Settings and Configuration

| File | Description |
|---|---|
| `Settings.ini` | Stores the QlikView Server (QVS) settings. Manual changes in this file require restart of QVS. This file is always stored in the "Data" folder (see *Overview (page 53)*). |

#### Cluster

QVS uses `.pgo` files to coordinate a cluster. The files are stored in the "Data" folder (see *Overview (page 53)*).

| File | Description |
|---|---|
| `BorrowedCalData.pgo` | Keeps track of borrowed Client Access Licenses (CALs). |
| `CalData.pgo` | Keeps track of CALs. |
| `IniData.pgo` | Coordinated version of `Settings.ini`. |
| `ServerCounters.pgo` | Keeps track of statistics. |
| `TicketData.pgo` | Keeps track of tickets. |

#### Logs

The logs are kept one per node in the cluster. The log files are stored in the "Data" folder by default (see *Overview (page 53)* for the default path).

| File | Description |
|---|---|
| `Events_<computer_name>.log` | Event log. |

| File | Description |
|---|---|
| `Performance_<computer_name>.log` | Performance log. |
| `Sessions_<computer_name>.log` | Session log. |

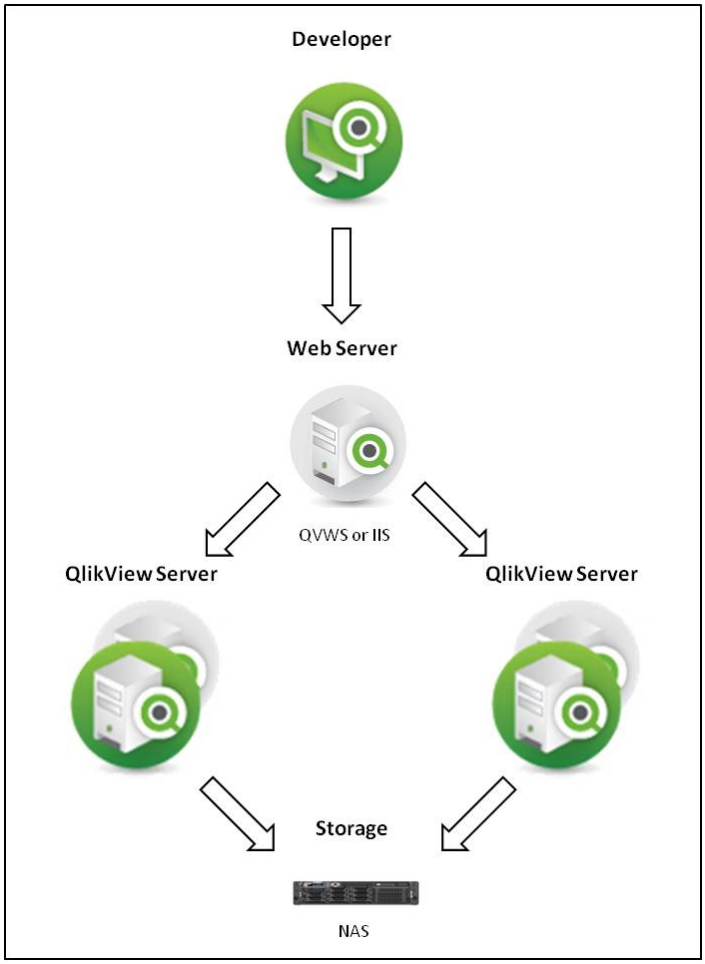
## Special Folders

The special folders are stored in the "Data" folder (see *Overview (page 53)* for the path).

| Folder | Description |
|---|---|
| `Extensions` | **Note!** The `Extensions` folder has to be created manually.<br>By default, QVS looks for extensions in this folder. Extension objects are located in `Extensions\Objects` and document extensions are located in `Extensions\Document`. Use QlikView Management Console (QMC) to manage all extensions in one place in case of a cluster. |
| `Temp` | By default, QVS puts temporary files in this folder (for example, when exporting using the AJAX client, a temporary file is created in the folder). |

# Load Sharing (Clustering)

All clustering requires a cluster-enabled QlikView Server license. QlikView Server supports load sharing of documents across multiple machines. This sharing includes the ability to share in real time, information about server objects, automated document loading, and user license CALs. Special licensing is available to enable multiple server instances share the same license number.




*Load sharing using QlikView Web Server*

To use load sharing between multiple QVSs, all document and support files must be shared between the servers. In other words, all servers should point to the same physical location for the files. QVS creates and maintains additional files to store load sharing data. These files have a Persistent Group Object (`.pgo`) file type extension and are located in the "Data" folder (see *Overview (page 53)*). These files are locked when QVS is running. The different `.pgo` files contain information on borrowed CALs, CALs in use, server settings, and ticket data.

Operating system load balance or failover configurations are external to the QVS load sharing configuration, and QVS has no control over those systems.

Server configuration settings are shared between all clustered QVSs and can be maintained through QMC connected to any of the clustered QVSs. Performance of a particular QVS system can be monitored through QMC by connecting to that system. The load balancing settings, that is, which QVS the client should be directed to, are stored in QlikView Web Server (QVWS).

Document-related meta data is shared via `.meta` files (one per document). This data is often referred to as Document Metadata Service (DMS) data. Since DMS data is shared among the QVSs, any automated document load procedures are performed on all servers. DMS authorization is also shared among all clustered QVSs.

# 9.2 QlikView Distribution Service

## Overview

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Directory Service Connector\QVDirectoryServiceConnector.exe` |
| **Data** | `%ProgramData%\QlikTech\DistributionService` |
| **Listens to** | HTTP: 4720; SNMP: 4721 |
| **Uses/Controls** | DSC, QVS, QVB |
| **Used by** | QMS |

**Note!** After restarting the machine, the Windows event log may contain a message that the QlikView Distribution Service (QDS) failed to start in a timely manner, even though it started successfully. This is because the QDS initialization phase is longer than the Windows timeout period (30 seconds by default). To avoid the event log message, either change the Windows timeout period or configure QDS to depend on another late starting service to make QDS start up during a less busy period.

## Files

The QlikView Distribution Service (QDS) files can be divided into three groups based on main purpose. All files are stored in the QDS "Data" folder (see *Overview (page 56)*). In a clustered setup, all QDSs must share the same program folder. This is solved by the file `config_<computer_name>.xml`, which contains the program data path to use.

### Settings and Configuration

The files listed below are local copies of the information stored in QVPR.

| File | Description |
|---|---|
| `Configuration.xml` | Configuration file for the service. |
| `Tasks\Task_<GUID>.xml` | The actual tasks. Note that deleted tasks are not automatically removed (due to support issue analysis). |
| `Triggers\Triggers_<GUID>.xml` | The actual triggers. Note that deleted triggers are not automatically removed (due to support issue analysis). |
| `Notification.xml` | Used to synchronize `Configuration.xml`, `TaskDetails.xm`, and `TriggerDetails.xml` with QVPR. |
| `TaskDetails.xml` | A list of the available tasks in the `Tasks` folder. In addition, used to synchronize the files in that folder with QVPR. |
| `TriggerDetails.xml` | A list of the available triggers in the `Triggers` folder. In addition, used to synchronize the files in that folder with QVPR. |

## Cluster

| File | Description |
|---|---|
| LoadBalancer.xml | Used to select which QDS (in a cluster) to do the job. |

## Logs

| File | Description |
|---|---|
| TaskResults\TaskResult_<GUID>.xml | Latest result of the task identified by the GUID. |
| TaskLogIndex\TaskLogIndex_<GUID>.xml | This is just for lookup (one file per task), pointing to the actual log. |
| EdxResults\EdxResult_<GUID>.xml | Until the task is completed, this file contains the current status of the EDX task. When the execution is finished, it contains the result (success/fail) and the task started as a result (if any). |
| <node-nr>\Log\<Date>.txt | General QDS event and error log. |
| <node-nr>\Log\Cluster_<Date>.txt | Synchronization log. |
| <node-nr>\Log\LoadBalancer_<Date>.txt | Load balancing log. |
| <node-nr>\Log\Root_<Date>.txt | QDS event log. |
| <node-nr>\Log\WebService_<Date>.txt | QDS event log. |
| <node-nr>\Log\Workorder_<Date>.txt | QDS event log. |
| <node-nr>\Log\<date>\<time> - <task name>\Tasklog.txt | QDS task event log. |
| <node-nr>\Log\<date>\<time> - <task name>\DistributionReport.xml | The distribution related to the task (only exists for distribution tasks). |

## QlikView Batch

### Overview

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Distribution Service\qvb.exe` |
| **Data** | - |
| **Listens to** | COM |
| **Uses/Controls** | - |
| **Used by** | QDS |

**Note!** QlikView Batch (QVB) does not support graphical or user input objects. This means that QVB cannot reload documents that, for example, contain scripts that require user input.

### Files

#### Settings and Configuration

| File | Description |
|---|---|
| `Settings.ini` | Used to store settings. |

#### Logs

| File | Description |
|---|---|
| `<document_name>.log` | Reload log that is placed together with the reloaded document. |

# 9.3 QlikView Publisher Repository

## Overview

| | |
|---|---|
| **Executable** | - |
| **Data** | `%ProgramData%\QlikTech\ManagementService\QVPR` |
| **Listens to** | - |
| **Uses/Controls** | - |
| **Used by** | QMS |

## Files

By default, QlikView Publisher Repository (QVPR) is a set of XML files. These files are backed up as `.zip` files in `%ProgramData%\QlikTech\ManagementService\QVPR\Backups`.

## Security Groups

When installing QlikView Server/Publisher, a couple of security groups are created.

The QlikView Server/Publisher services must run under an account that is member of the security group QlikView Administrators. Users connecting to QMC must be part of this group. Anyone connecting to a remote service must also be member of QlikView Administrators.

The users connecting through the API must be members of the QlikView Management API security group. The group is not created during the installation and has to be added (and populated, for example, with the members of the QlikView Administrators group) manually. A membership in this group is required to import tasks from another QlikView Server/Publisher.

The QlikView EDX security group is not created during the installation and has to be added (and populated) manually in order for users to run EDX tasks.

## Document Administrators

To delegate the responsibility of creating tasks to people not part of the QlikView Administrators group, users can be appointed document administrators. The document administrators are only allowed to access the tabs in QMC that are related to either user documents or source documents.

**Note!** The use of document administrators requires a QlikView Publisher license.

For more information on how to appoint document administrators, see the QMC online help.

# 9.4 Configuration Files

**Note!** Use QMC to set the parameters described in this section, since modifying the configuration files directly may cause problems.

## Management Service – QVManagementService.exe.config

In a default installation, this file is located in `%ProgramFiles%\QlikView\Management Service`. The file has a number of automatically generated tags that should not be modified, but the settings listed below can be modified.

| Setting | Description |
|---|---|
| ApplicationDataFolder | Folder where the log folder and all other files/folders are created. The default value is `%ProgramData%\QlikTech\ManagementService`. This folder is where the XML version of QVPR and the LEF information are stored. |
| UseHTTPS | `True` = Communication runs over https. A certificate for the web site is needed to enable this setting. |
| Trace | Used for debug logging. |
| QMSBackendWebServicePort | Port that the back end management service listens to. The default value is 4799. |
| QMSFrontendWebServicePort | Port that the front end management service listens to. The default value is 4780. |
| MaxLogRecords | Maximum number of log records that should be retrieved for a task. |
| EnableAuditLogging | `True` = Track a) changes on tasks and settings made in the system, b) who made the changes, and c) when the changes were made. |
| AuditLogFolder | Path to the folder where the audit logs are saved. |
| AuditLogKeepMaxDays | Maximum number of days each log is saved. |

For additional information, see *SNMP (page 125)*.

## Distribution Service – QVDistributionService.exe.config

In a default installation, this file is located in `%ProgramFiles%\QlikView\Distribution Service`. The app settings tag is the part that can be modified. Some of the settings in the configuration file are described below.

| Setting | Description |
|---|---|
| ApplicationDataFolder | Folder where the log folder and all other files/folders are created. The default value is `%ProgramData%\QlikTech\DistributionService`. This folder is where the XML version of QVPR and the LEF information are stored. |
| WebservicePort | Port that the QlikView Distribution Service uses to communicate with. The default value is 4720. |
| UseHTTPS | `True` = Communication runs over https. |
| DSCAddress | Port that the Directory Service Connector service uses to communicate with. The default value is 4730. If the value is modified, the tag "DSCAddress" in the `QVDirectoryServiceConnector.exe.config` file has to be modified too. |
| DSCTimeoutSeconds | Timeout for calls to the Directory Service Connector. |
| DSCCacheSeconds | How long the service caches the responses from the Directory Service Connector. |
| QlikViewEngineQuarantineTimeInms | How often a QlikView engine is allowed to start (in milliseconds). |
| OpenDocumentAttempts | How many tries that can be made to open a document before it is logged as an error during distribution. |
| DebugLog | `True` = Enable logging of memory usage and stack trace on "Error" logging. |
| Trace | `True` = Enable debug logging. |
| EnableBatchMode | Enable this setting to make batch calls to the QlikView Distribution Service (see *QlikView Distribution Service (page 56)* for more information). |

For additional information, see *SNMP (page 125)*.

### Directory Service Connector – QVDirectorServiceConnector.exe.config

This file is by default located in `%ProgramFiles%\QlikView\Directory Service Connector\QVDirectoryServiceConnector.exe.config`. The settings most commonly modified are listed below.

| Setting | Description |
|---|---|
| ApplicationDataFolder | Folder where the log folder and all other files/folders are created. The default value is `%ProgramData%\QlikTech\DirectoryServiceConnector` (`C:\Documents and Settings\All Users\Application Data\QlikTech\DirectoryServiceConnector` on pre-Windows Vista systems). |
| WebservicePort | Port that the Directory Service Connector service uses to communicate with. The default value is 4730. If the value is modified, the tag "DSCAddress" in the `QVDistributionService.exe.config` file has to be modified too. |
| UseHTTPS | `True` = Communication runs over SSL instead of http. A certificate for the web site is needed to enable this setting. |
| PluginPath | Path where the Directory Service Connector looks for available DSP plugins. The default value is `%ProgramFiles%\QlikView\Directory Service Connector\DSPlugins`. |
| Trace | `True` = Enable debug logging. |
| DisableCompress | Enable this setting to disable compression of the http communication. |

For additional information, see *SNMP (page 125)*.

## 9.5 Web Server

The web server can be the built-in QlikView Web Server (QVWS) or Microsoft IIS. QVWS is installed as a Windows service during a default, complete installation of QlikView Server. When IIS is used, the same functionality is provided by a set of ASPX pages and a special support service, QlikView Settings Service (QSS). QSS acts as the management interface for settings used by the ASPX pages.

### Overview

#### QlikView Web Server

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Server\Web Server\QVWebServer.exe` |
| **Data** | `%ProgramData%\QlikTech\WebServer` |
| **Listens to** | HTTP: 80; HTTP: 4750; SNMP: 4751 |
| **Uses/Controls** | DSC |
| **Used by** | Web browser clients and mobile clients |

### QlikView Settings Service

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Server\Web Server Settings\QVWebServerSettingsService.exe` |
| **Data** | `%ProgramData%\QlikTech\WebServer` |
| **Listens to** | HTTP: 4750 |
| **Used by** | QMS |

## Files

### Settings and Configuration

| File | Description |
|---|---|
| `Config.xml` | Configuration file for the service. |

### Logs

| File | Description |
|---|---|
| `Log\<date>.txt` | Event and error log. |

## Load Balancing

QVWS hosts web pages, prepares the file list for AccessPoint, and manages the load balancing of QlikView Servers (QVSs).

AccessPoint is a web portal for documents hosted on QVWS. The pages for AccessPoint are by default located in the folder `%ProgramFiles%\QlikView\Web`. QVWS also acts as web server for any AJAX pages accessed by the end users.

The load balancing performed by QVWS is different from load balancing a web server, since the additional work and resource consumption is almost similar for each user, so it does not matter on which server the user ends up.

The load balancing schemes are listed below.

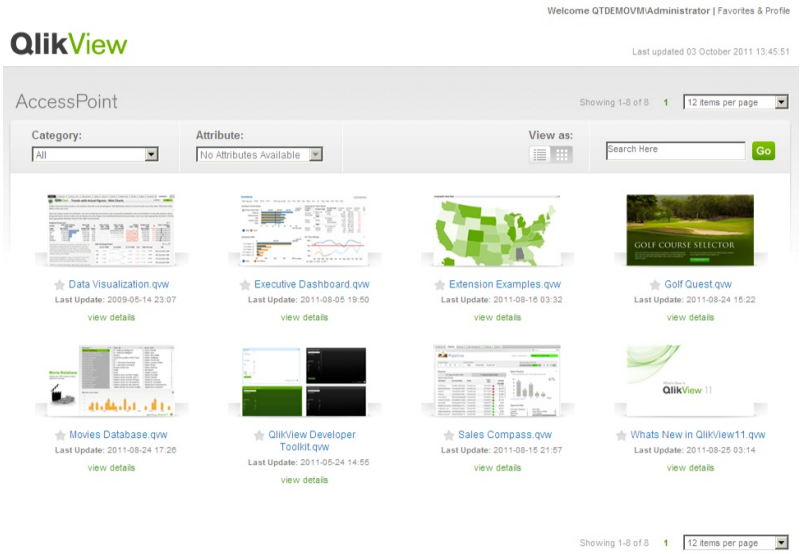
| Scheme | Description |
|---|---|
| Random | The default load balancing scheme. The user is sent to a random server, no matter if the document the user is looking for is loaded or not. |
| Loaded Document | If only one QVS has the particular document loaded, the user is sent to that QVS. If more than one QVS or none of the QVSs has the document loaded, the user is sent to the QVS with the largest amount of free RAM. |
| CPU with RAM Overload | The user is sent to the least busy QVS. |

The settings for load balancing are configured in QMC.

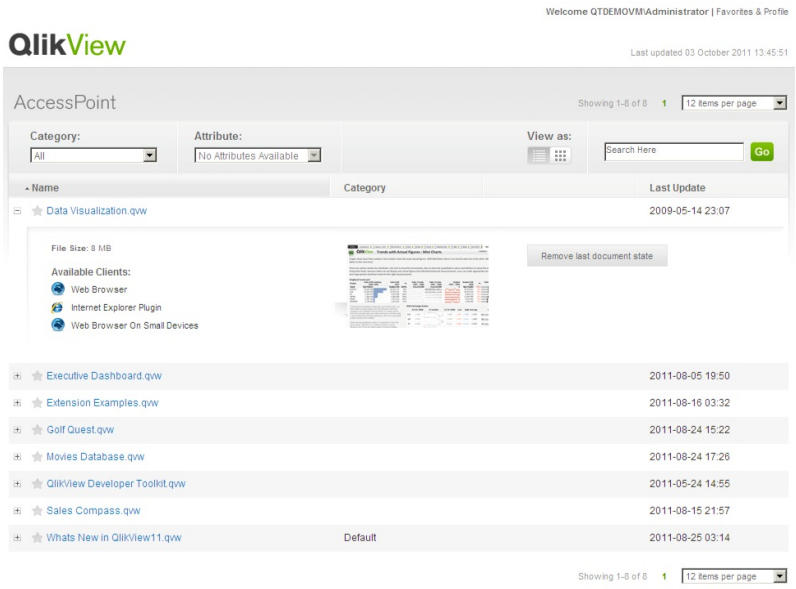
# QlikView AccessPoint

QlikView AccessPoint is a web portal that lists the documents each user has access to. AccessPoint only links to each document – it does not host the documents. The hosting is done by QlikView Server.

The documents can be displayed as thumbnails or in a detailed list.




*Thumbnails view in AccessPoint*




*Detailed view in AccessPoint*

The settings available in AccessPoint are listed below.

| Setting | Description |
|---|---|
| Category | Category grouping for the document. Categories are managed in QMC under **Documents>User Documents>Document Information**. |

| Setting | Description |
|---|---|
| Attribute | Attribute grouping for the document. Attributes are managed in QMC under **Documents**>**User Documents**>**Document Information**. |
| View as | Document display type, **Detailed** view or **Thumbnails** view.<br>In the Detailed view, the documents can be sorted by Name, Category, and Last Update. |

Click a **view details** link in the Thumbnails view or a plus sign (+) to the left of a document name in the Detailed view to display additional information on a document (see below).

| Field/Button | Description |
|---|---|
| Last Update | When the document was last updated.<br>**Note!** This is only displayed in the Thumbnails view. |
| Next Update | When the document will be updated next time.<br>**Note!** This is only displayed if the document is part of a task that has a schema. |
| File Size | Size of the document. |
| Available Clients | Click a client to open the document with that client. |
| Remove last document state | Click this button to remove the last document state. |

Click a star icon next to a document name in the Thumbnails or Detailed view to set the preferences for the document.

| Setting | Description |
|---|---|
| Open with | Select a client to make it the default client to open the document with. |
| Add to favorites | Click this link to add the document to the favorite documents. Select **Category**>**Favorites** in AccessPoint to display the favorites. |

# 9.6 Directory Service Connector

For information on the Directory Service Provider (DSP) interface, see *DSP Interface (page 123)*.

## Overview

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Directory Service Connector\QVDirectoryServiceConnector.exe` |
| **Data** | `%ProgramData%\QlikTech\DirectoryServiceConnector` |
| **Listens to** | HTTP: 4730; SNMP: 4731 |
| **Uses/Controls** | - |
| **Used by** | QDS, QMS, QVWS |

## Files

### Settings and Configuration

These settings originate from QVPR.

| File | Description |
|---|---|
| `Config.xml` | Configuration file for the service. |
| `Resources/<id>.xml` | DSP configurations. |

### Logs

| File | Description |
|---|---|
| `Log\<date>.txt` | Event and error log. |

# 9.7 QlikView Management Service

## Overview

| | |
|---|---|
| **Executable** | `%ProgramFiles%\QlikView\Management Service\QVManagementService.exe` |
| **Data** | `%ProgramData%\QlikTech\ManagementService` |
| **Listens to** | HTTP: 4780 (Web); HTTP: 4799 (API); SNMP: 4781 |
| **Uses/Controls** | DSC, QDS, QVS, QVWS |
| **Used by** | Web browser/API client |

## Files

### Settings and Configuration

QlikView Management Service (QMS) keeps a global view of the settings in QVPR.

| File | Description |
|---|---|
| `Config.xml` | Configuration file for the service. |

### Logs

| File | Description |
|---|---|
| `Log\<date>.txt` | Event and error log. |

# Part 4 Security

# 10 Security Overview

The security of QlikView Server/Publisher consists of the following parts:

- Protection of the platform: How the platform itself is protected and how it needs to communicate and operate.
- Authentication: Who is the user and how can the user prove it? QlikView uses standard authentication protocols, such as Integrated Windows Authentication (IWA), HTTP headers, and ticketing, to authenticate every user requesting access to data.
- Document level authorization: Is the user allowed to access the document or not? QlikView uses server-side capabilities such as Document Metadata Service (DMS) or Windows NTFS to determine access privileges at file level.
- Data level authorization: Is the user allowed to see all of the data or just parts of it? QlikView implements row and field level data security, using a combination of document-level capabilities (Section Access) and server-side data reduction capabilities (QlikView Publisher).

# 11 Protection of the Platform

## 11.1 Functionality

The functionality for downloading documents and/or print and export to Microsoft Excel can be restricted at the user level for each document on the server.

## 11.2 Special Accounts

### Supervision Account

The supervision account is granted access to all documents that are created by tasks in QlikView Publisher. The characteristics of the supervision account are as follows:

- Provides access to *all* files on the QVS
- Does *not* provide any access to the QlikView Management Console (QMC)
- Respects the types of clients that are allowed for each document (for example, a supervision account cannot open a QlikView document using the AJAX client, if the AJAX client has been blocked by the user that created the task)

### Anonymous User Account

When QVS is started for the first time on a machine, a Windows account is created for anonymous users. The account name is `IQVS_name`, where `name` is the name of the machine in the local network.

If the machine in question is a domain server, the anonymous account is created as a domain account. If not, it is created as a local machine account.

Each folder and file that is to be available for anonymous clients must be given read privileges for the anonymous account.

**Note!** Start QVS and let it create the anonymous account before attempting to grant any privileges. Do not try to create the anonymous account manually.

### QlikView Administrators

The QlikView Administrators group is used for granting access to the QlikView Management Console (QMC) as well as authorization of communication between services, if Windows Authentication is used.

## 11.3 Communication

### Protection of AJAX Client

The AJAX client uses HTTP or HTTPS as the protocol for communication between the client browser and the QlikView Web Server (QVWS) or Microsoft IIS. It is strongly recommended to protect the communication between the browser and the web server using SSL/TSL encryption over the HTTP protocol (that is, HTTPS). If the communication is not encrypted, it is sent as clear text.

The communication between the web server and QVS uses QVP as described below.

### Protection of Plugin

The QlikView plugin can communicate with QVS in two ways. If the plugin has the ability to communicate with QVS using QVP (port 4747), the security described in *Server Communication (page 72)* is applied.

If the communication cannot use QVP or if the client chooses it in the plugin, the communication is tunneled using HTTP to the web server.

If HTTPS is enabled on the web server, the tunnel is encrypted using SSL/TLS.

## Server Communication

The QVS communication uses the QVP protocol, which is encrypted by default. The QVP protocol can be protected using 1024-bit RSA for key exchange and 128-bit RC4 for data encryption, provided the Microsoft Enhanced Cryptographic Provider is installed. If the Microsoft Base Cryptographic Provider is used, the protection of the communication is 512-bit RSA for key exchange and 40-bit RC4 for data encryption.

## Services Communication

The services that are part of the QlikView platform (that is, QVS, DSC, QMC, QDS, and QVWS) all communicate using web services. The web services authenticate using Integrated Windows Authentication (IWA).

# 12 Authentication

Although QlikView can be configured to allow anonymous access, the majority of implementations require users to be authenticated. In such environments, QlikView always requires that the user is authenticated when establishing a session via QlikView Server (either through a browser or when downloading and opening a document via the QlikView Desktop client).

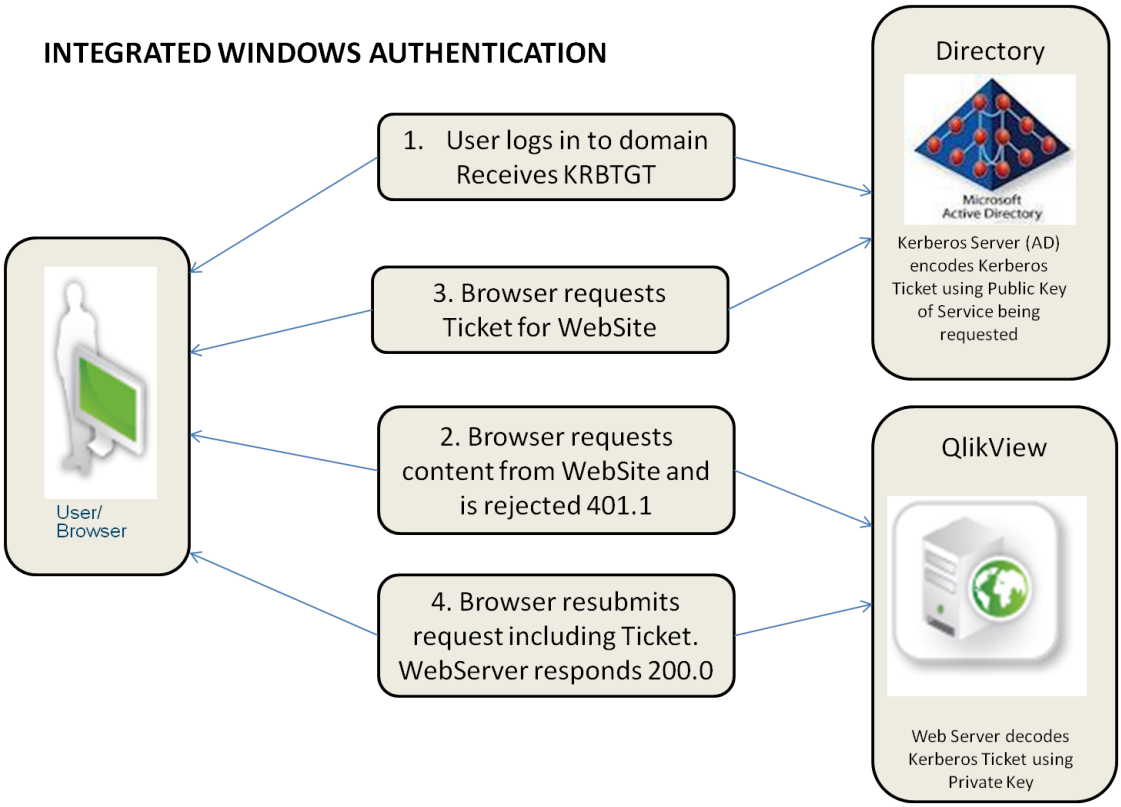
In the QlikView context, the authentication of a user is almost always done against an external entity that is then used to pass the externally authenticated user identity to QlikView Server. In such a scenario, QlikView relies on the authentication to be performed prior to accessing QlikView, and that some token of identity is transmitted to, and trusted by, QlikView.

## 12.1 Authentication when Using QlikView Server in a Windows User Environment

Authentication to a QlikView Server in an environment based on Windows users (for example, incorporating Active Directory) is straightforward. The process is as follows:

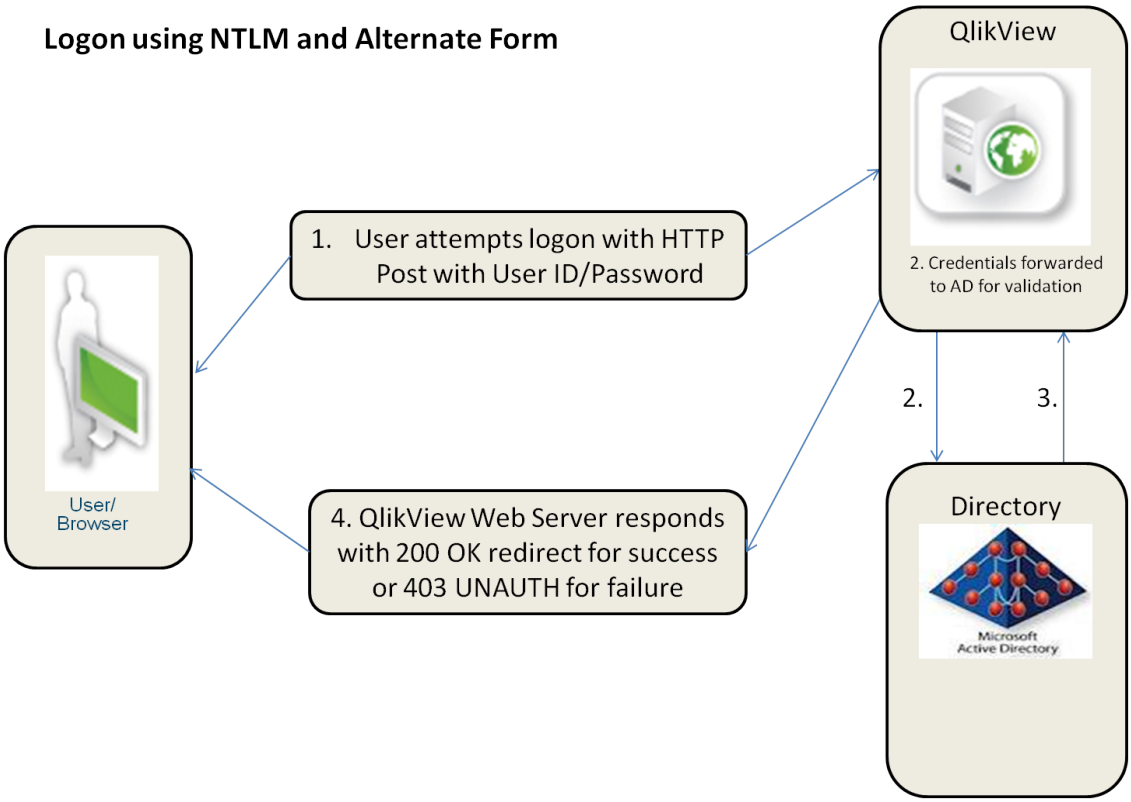
1. The user credentials are validated when the user logs in to the Windows operating system on the client machine.
2. Later when the user wants to establish a session with a QlikView Server (QVS) (for example, via a browser on the desktop), QVS can use the built-in Integrated Windows Authentication (IWA).
3. The identity of the logged-in user is communicated to QlikView Server using either the Kerberos or the NTLM security solution. This solution provides single sign-on capabilities right out of the box. In case the authentication exchange fails to identify the user, the browser prompts the user for a Windows user account name and password.

The figure below shows the standard authentication flow for IWA:




*Authentication when using QlikView Server in a Windows user environment*

The figure below shows the authentication flow for the combination of NTLM and alternate login, which differs from the standard flow for IWA:




*Authentication using NTLM and alternate form*

The authentication process differs based on the environment:

- Local Area Network (LAN): IWA is most common and most suitable for recognizing Windows users on a LAN. The act of authentication is performed when logging in the workstation, and this identity is leveraged by QlikView.
- Multi-domain environment: The internal company network IWA should be avoided in architectures where there is a multi-domain environment with no trust relationship between the domain of the workstation and the domain of the server, or when used across a reverse proxy. In such an environment, configure the QlikView deployment to use either an existing external SSO service or a QlikView custom ticket exchange to expose an authenticated identity to QlikView.

## 12.2 Authentication with a QlikView Server Using an Existing Single Sign-on Software Package

In environments where an SSO infrastructure already exists (for example, CA SiteMinder®, IBM® WebSeal, or Oracle® Oblix), QlikView can use the HTTP header injection method of single sign-on provided by the SSO infrastructure. This means single sign-on is provided right out of the box. The SSO infrastructure software packages can be configured as follows:

- Repeat user get access: The software packages can be configured to protect a resource. When a user requests access to QlikView, the SSO package grants access, if the user has previously signed in to the SSO authentication page.
- New user log in: If the user does not have an existing session with the SSO package, the user is redirected to the SSO package login page. After logging in, the user is redirected to the original URL that the user requested.

In both cases, if the user has properly authenticated to the SSO software, the username is injected into an HTTP header and the value in that header is what the QlikView server accepts as the authenticated identity of the user.

**Note!** Unless SSO software is in place, the HTTP header method of authenticating to a QlikView Server must not be used. HTTP headers can easily be spoofed. All of the SSO software packages mentioned above provide protection against this type of spoofing attacks, if the software package is the only path for users to access the content.

QlikView does not recommend or endorse any specific tool or product for providing identity in HTTP headers. The approach is highly suited to extranet deployments wherein the users may not exist in the internal Active Directory. The act of authentication is performed by the reverse proxy or ISAPI filter that intercepts the attempt of the end user to interact with QlikView content.

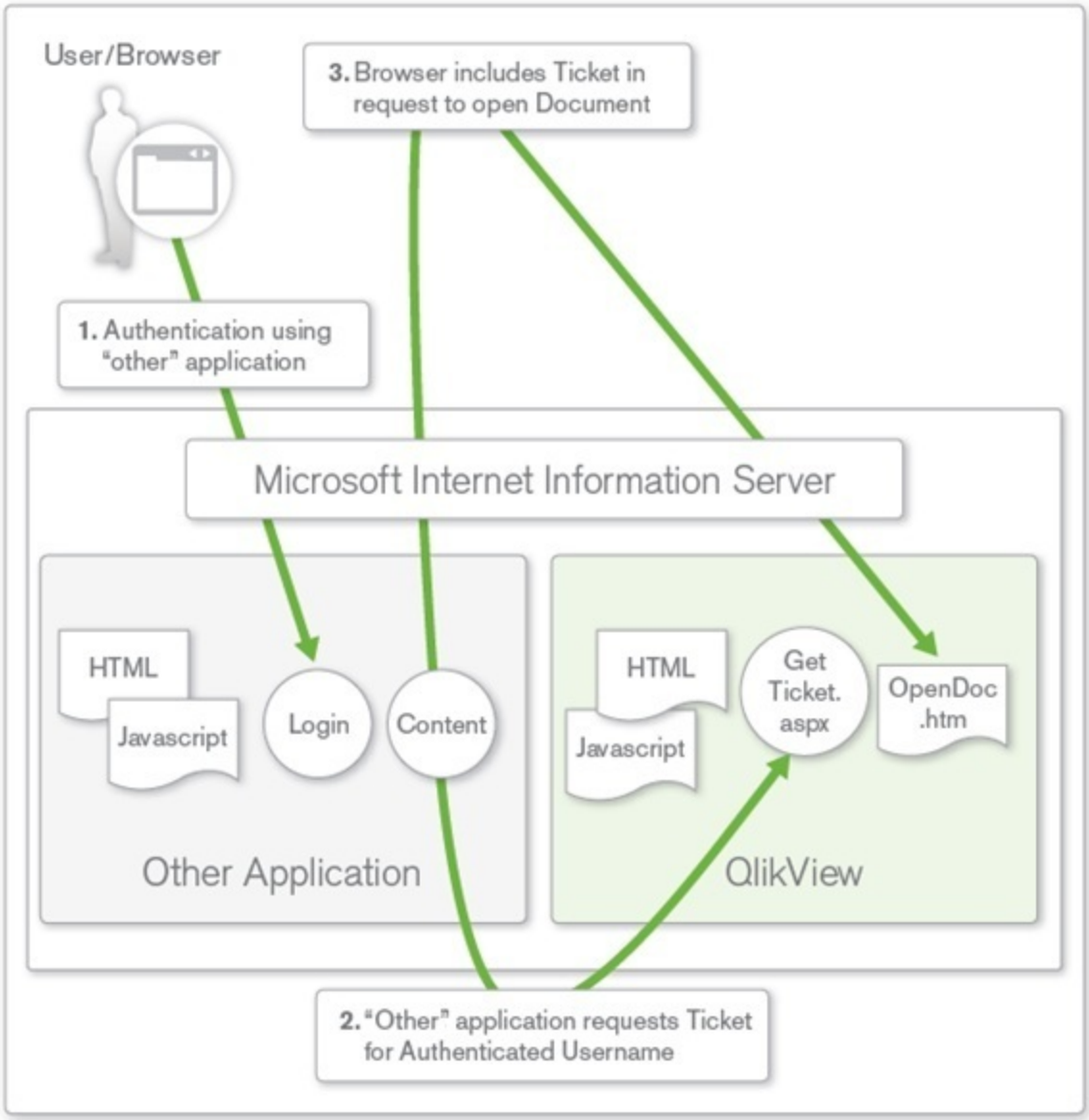
## 12.3 Authentication Using neither IWA nor Single Sign-on Software

QlikView provides a third method for single sign-on, Custom Ticket Exchange (CTE), when neither of the methods described above is suitable.

CTE relies on the user having authenticated previously to another system:

1. The third-party system is granted the privilege and responsibility to request an authentication token (called a “ticket” in QlikView) from QVS on behalf of the authenticated user of the third-party system. It is the responsibility of the third-party system to only request tickets for users that have been properly authenticated (for example, QVS has no knowledge of the authentication status of the user).
2. The system then passes the authentication token to the user, who uses it in a request to open a session with QVS.
3. QVS checks that the ticket is valid and then opens a session for the authenticated user.

Ticketed authentication is mainly applicable when embedding QlikView content in third-party applications and portals, and is rarely used for providing general access to QlikView. Typically a small amount of custom development is needed to implement the request and passing of the ticket for the CTE method to work.

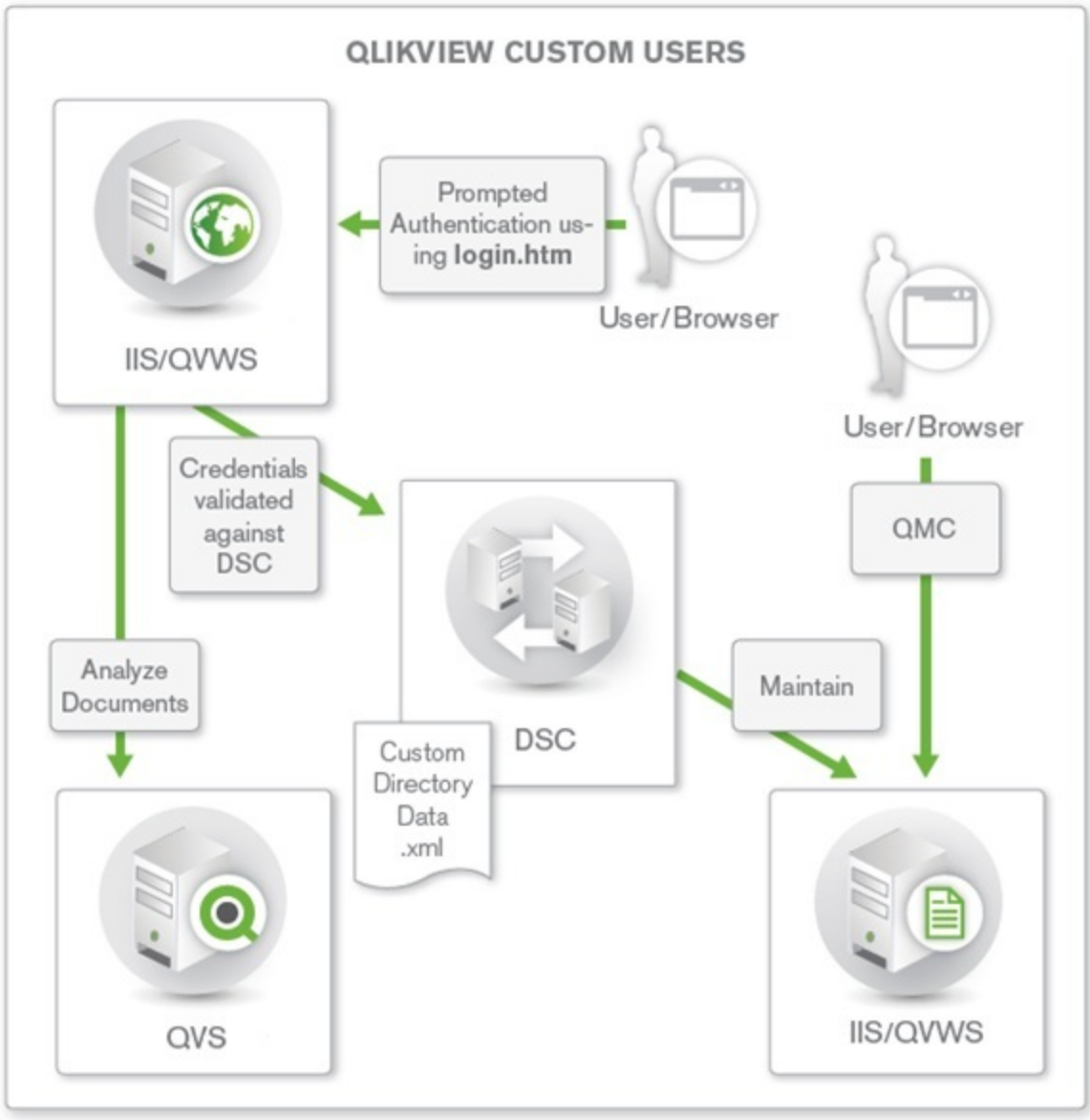



*Authentication using neither IWA nor single sign-on software*

# 12.4 QlikView Server Authentication Using Custom Users

The three methods described above all use a single sign-on principle, where the user ID and password are stored externally to QlikView Server and an external entity is responsible for the authentication. Less common, although possible, is the ability to store the user credentials in the QlikView Server environment using the Custom Users functionality in QlikView Publisher. In this case, users and passwords are defined and stored within the QlikView environment and the web tier of the QlikView deployment is responsible for forms authentication. This solution is suitable for smaller, standalone QlikView Server deployments, and must not be used in environments where the user definitions are to be available to multiple systems. In such environments, it is highly recommended to use one of the three single sign-on solutions described above.

Each coexistent form of authentication may require a distinct web server instance. Several web servers can forward user requests to the same QVS instance(s).




*QlikView Server authentication using Custom Users*

# 13 Authorization

Once a user has been authenticated (that is, the system knows who the user is), the first step in assigning the security privileges has been completed. The second step is to understand the authority or access rights that the user has to applications, data, or both. This step is referred to as Authorization. At a fundamental level, an administrator populates an Access Control List (ACL) with a list of users and/or groups and what they are to have access to. When the time comes for a user to request access, the system looks up the authenticated identity of the user in the ACL and verifies if the administrator has granted the user enough privileges to do so.

Direct access to a QlikView document using QlikView Desktop is always governed by the Windows NTFS file security. Access to the web-based QlikView Management Console (QMC) is restricted to Windows users that are members of a particular local Windows group.

## 13.1 Document Level Authorization

Once a user has been authenticated, QlikView Server typically handles authorization on its own. QlikView Server provides the choice between storing the ACL information as Windows NTFS privileges (applicable only when the user is authenticated using a Windows user identity) or by storing the ACL information in the internal repository, Document Metadata Service (DMS), in QlikView. The choice of NTFS or DMS affects the access to all documents in QlikView Server.

### NTFS vs. DMS

QlikView Server can use the NTFS privileges of the Windows file system to store authorization information. When in NTFS authorization mode, QlikView Server controls access to a given QlikView document by determining if the authenticated user has NTFS privileges to the underlying QlikView document file (`.qvw`). This is based on the operating system privileges and Windows NTFS is used for the ACL. The privileges of the authenticated user are configured by a server administrator using standard Windows Explorer functionality via directory properties options.

As an alternative to Windows NTFS, QlikView can use its own ACL, DMS. Unlike NTFS, this allows non-Windows users and groups to be authorized to access applications and data. DMS integrates fully with the existing Directory Service Provider (for example, Active Directory, other LDAP) where Group Membership has been recorded – this is a mechanism by which QlikView Server can re-use existing enterprise accounts and group structures. The permitted users or groups are recorded in a meta file that resides next to the QlikView document, and it is managed using QMC.

NTFS is the default document authorization model, suitable when all users and groups are identified in Active Directory or locally on the QlikView Server host. The NTFS permissions may be inherited from the directory that the QlikView documents are in, or may be assigned using QlikView Publisher distribution tasks.

DMS is required when the authenticated user identity is not a Windows user account. The DMS permissions are explicitly assigned using QMC, or may be assigned using QlikView Publisher distribution tasks.

**Note!** When authenticating a user via a web ticket, the user is not a proper Windows user, even if sending in the user name in Active Directory format (for example, QLIKVIEW\jsmith). This means that DMS authorization should be used when using web tickets.

## 13.2 Data Level Authorization

Data level authorization allows access to be granted or denied on a document level and even to specific data in a document.

There are two types of data level authorizations:

- Dynamic data reduction: Determines if the user is allowed to view the data when the user tries to access it.
- Static data reduction: Performed by QlikView Publisher, determines if the user is allowed to view the data when it is prepared for the user.

Static and dynamic reduction of data can be used on its own, but can also be combined to deliver data level authorization.

# Dynamic Data Reduction

Dynamic data reduction is done in QlikView using the concept of Section Access, which is part of the QlikView document.

Section Access Management is configured in the QlikView Management Console (QMC). For information, see the QMC help.

# Static Data Reduction

For larger deployments and/or those in need of centralized control of authorization capabilities, QlikView Server/Publisher are used. Departments or functions often have a "master" application that contains all relevant data covering all analysis needs, and this master document needs to be separated ("reduced") according to the needs and access privileges of the intended audience. QlikView Publisher reloads the QlikView document with available data, refreshes the Section Access tables, and splits the large QlikView document into smaller documents based on values in a particular field.

This "reduction and distribution" allows for a file containing many data fields to be broken up by the contents of a field and distributed to authorized users or groups according to their access privileges.

One of the benefits of reducing and distributing source files in this manner is that the documents that are created in this process contain no explicit reference to the source data (for example, a database connection string) in their script environments. Therefore, if a user interacts with the document via QlikView Desktop, the user cannot see the location of the source data. All of the data pertinent to the user needs is contained in the document.

An administrator can use QMC to create tasks on source `.qvw` or `.qvd` files to accomplish this. At a basic level, the steps are as follows:

1. On the source document (either `.qvw` or `.qvd`), apply the data reduction criteria (for example, choose the field name on which to reduce the data).
2. Apply the distribution criteria to the newly created (reduced) files:
   a. Assign the authorization privileges using either DMS or NTFS ACLs.
   b. Choose the type of distribution (for example, `.qvw` files or `.pdf` report).
   c. Choose the location for the newly created files.
3. Apply the notification criteria for the completion of the task (for example, e-mail notification)

The newly created files only contain the data that the user or group is authorized to see, since the data has been "reduced" from the master document in accordance to the reduction criteria. This is why the process is termed "Static Data Reduction". Hence, there is no risk of an unauthorized person viewing data, since only authorized data exists in each file.

# Part 5 Licensing

# 14 Client Access Licenses

To connect to a QlikView Server (QVS), each client needs a Client Access License (CAL). The CALs are purchased with QlikView Server and tied to the server serial number. A CAL is never transferred to a client, but a client uses the CAL when connecting to the server or, if a cluster license is used, a specific QlikView Server cluster. CALs cannot be transferred between different QlikView Server clusters. If a user is required to work with documents residing in different clusters, a separate CAL is needed for each of the clusters.

**Note!** The CALs require the QlikView Servers within a cluster to be within the same geographical and physical location and within the same network sub-net/segment.

## 14.1 CAL Types

The CALs described below are used to provide users access to the various QlikView Servers identified in *Editions of QlikView Server (page 87)*.

**Note!** CALs are used for licensing only and they have nothing to do with user authentication for data access purposes.

| CAL Type | Description |
|---|---|
| Named User CAL (an identified user on a server) | A Named User CAL is assigned to a unique and identified user (see *Identification (page 84)* for information on how users are identified) who may access as many QlikView documents as may reside on the server or server cluster to which the Named User CAL is assigned. A Named User CAL may be transferred to another user pursuant to the software licensing agreement, in which case there is a 24-hour quarantine before the Named User CAL can be transferred to another user. There is no time limit for how long a user assigned a Named User CAL can access a QlikView document. |
| Document CAL (an identified user within a given document) | A Document CAL is assigned to a unique and identified user (see *Identification (page 84)* for information on how users are identified) who may access only the one QlikView document to which the Document CAL is assigned. Multiple Document CALs can be assigned to a particular user. For example, if a user connects to two QlikView documents, the user will have been assigned two Document CALs. A Document CAL may be transferred to another user pursuant to the software licensing agreement, in which case there is a 24-hour quarantine before the Document CAL can be transferred to another user. There is no time limit for how long a user assigned a Document CAL can access the QlikView document to which the CAL is assigned. |
| Session CAL | A Session CAL allows any user, identified or anonymous/unidentified, on one QlikView client to access as many QlikView documents as may reside on the server or server cluster to which the Session CAL is assigned for a minimum period of 15 minutes. For Session CALs, the QlikView client refers to each unique instance of the QlikView client (for example, the AJAX client, QlikView Desktop, or the Internet Explorer plugin) on the user's machine. The minimum session time for a Session CAL is 15 minutes, which means that sessions that end in less than 15 minutes will still consume the session until the 15 minute mark is passed; those which terminate after 15 minutes will consume their actual session length. By default, there is no maximum session length, but this can be configured. |

| CAL Type | Description |
|---|---|
| Usage CAL | A Usage CAL allows any user, identified or anonymous/unidentified, to access only one QlikView document, residing on the server or server cluster to which the Usage CAL is assigned, from one client (for example, the AJAX client, QlikView Desktop, or the Internet Explorer plugin) for a time period of 60 minutes per 28-day period. If a user exceeds the 60 minute time limitation, the user will have consumed two Usage CALs without any warning being given to the user. Every 28 days, the Usage CAL is refreshed and the user may once again view a new QlikView document for 60 minutes, using the same Usage CAL. Usage CALs are continuously recharged (at a pace corresponding to 1/28 of the *total* number of Usage CALs assigned to the QlikView Server per day). |

## 14.2 Identification

To use a Named User CAL or a Document CAL, the client user must be identified via an authenticated user name (Windows Active Directory or through a ticket exchange between the web server and QlikView Server). An IP address is not a valid form of identification for a Named User CAL. The two methods of identification cannot be mixed on the same instance of QlikView Server. Note that the user name identification requires Windows authentication on AJAX clients, since machine name identification cannot be used for these clients.

Any CAL used by an identified user may not be transferred to another user, unless the transfer is due to a change in the employment status or work duties of the prior user, in which case there is a 24-hour quarantine before the CAL be transferred to another user.

## 14.3 Document CAL Restrictions

The purpose of the Document CAL is to provide a mechanism by which licensees can license the use of a single document. To prevent the combination of many data models in a single document, there are restrictions in the documents that can be used with the Document CAL. However, the Named User CAL, the Session CAL, and the Usage CAL can be used to open any functional QlikView document. The Document CAL can only be used with documents that have a single contiguous data model and do not contain any chasm traps between tables.

Most common data models used in QlikView documents can be used for Document CALs. For instance, proper star schemas and snowflake schemas typically have the field with the highest cardinality in the fact table and the keys in dimensional tables have a lower cardinality. For snowflake schemas, the cardinality decreases further when moving away from the fact table. Documents containing such models typically fulfill the above demands and are well-suited for Document CALs.

Documents with multiple logical islands are normally not allowed. Multiple logical islands are only allowed, if the additional tables are unconnected and contain only few records or a single column.

In addition, the document may not contain any loosely coupled tables.

Finally, the cardinality (that is, the number of distinct values) of the key fields must decrease when moving away from the fact table.

## 14.4 Combining Different CALs

A given instance of QlikView Server can carry any combination of the CAL types listed in *CAL Types (page 83)*. When different CAL types are combined on the same server, the order of priority in the CAL assignment is done as follows:

1. If there is a dedicated Named User CAL for the connecting client, it is used.
2. If there is a dedicated Document CAL for the connecting client, it is used.
3. If a new Named User CAL can be assigned to the connecting client, it is used.

4. If a new Document CAL can be assigned to the connecting client, it is used.
5. If there is an available Session CAL, it is used.
6. If there is an available Usage CAL, it is used.
7. If none of the above, access is denied.

## 14.5 License Lease

A QlikView client that does not have a registered license is allowed to connect to QlikView Server and "borrow" a license, so that the user can work offline for a period of 30 days. The QlikView client must then make an authenticated log on (not anonymous) and obtain a Named User CAL. Each time QlikView is started, QlikView tries to contact QlikView Server and renew the license lease. If the client cannot reach the server after 30 days, the license lease expires.

A license lease can only be used with QlikView Desktop and the Internet Explorer plugin. This means a license lease cannot be obtained when using an AJAX client.

## 14.6 Cluster Licensing

A special type of license is available to allow multiple QlikView Server installations to share the same license serial number and support shared CALs. The servers are automatically considered as clustered. Note that this configuration affects networks where unauthorized license sharing between test and production environments has been configured.

**Note!** The CALs require the QlikView Servers within a cluster to be within the same geographical and physical location and within the same network sub-net/segment.

## 14.7 Cold Standby Servers

QlikView Server license keys can be installed on as many servers as required provided that only the licensed number of QlikView Servers are running at any given time. Thus, a cold standby environment can be installed and ready-to-run, but cannot be live (that is, the Windows services cannot be started) and in use prior to the live environment being shut down.

# 15 Editions of QlikView Server

QlikView Server comes in a number of editions designed for different organizations and purposes. Upgrading is done through the license key.

## 15.1 Editions

The various editions of QlikView Server are listed below.

| Edition | Description |
|---|---|
| QlikView Enterprise Edition (EE) Server | QlikView EE Server is designed to be used in large and complex deployments and provides features such as unlimited documents, server-based collaboration, integration with third-party security systems, server clustering, and cluster licensing. The minimum configuration of a QlikView EE Server is five Named User Client Access Licenses (CALs). |
| QlikView Small Business Edition (SBE) Server | QlikView SBE Server is designed to be used in smaller deployments. The minimum configuration of a QlikView SBE Server is five Named User CALs. |
| QlikView Information Access Server (IAS) | QlikView IAS is a QlikView Server that is licensed for an uncapped number of users, but limited to a single QlikView document. QlikView IAS runs in anonymous mode and must be publically accessible without authentication (on the public Internet), that is, it must not be placed behind a firewall. No QlikView client can access QlikView IAS – the user interface must be built by the end user either manually or by using QlikView WorkBench.<br>**Note!** There is no license lease from QlikView IAS. |
| QlikView Extranet Server (QES) | QES allows end users to deploy QlikView solutions to their extranet. QES is based on QlikView EE Server, but only supports up to three QlikView documents. The server can be deployed with a combination of Session and Usage CALs. QES supports mobile clients and can be deployed in clustered environments. The AJAX client or a customized AJAX client can be used via QlikView WorkBench, which is included in QES. The minimum configuration of a QlikView QES Server is five Extranet Session CALs. |

In addition to the editions of QlikView Server described above, there is also a number of additional, server-related products, all of which are listed below.

| Product | Description |
|---|---|
| QlikView Test Server | QlikView Test Server is a license that provides an environment separate from production to use for data validation, application testing, and preparation/migration of QlikView documents to new versions and/or releases of QlikView.<br>QlikView Test Server comes in two editions, QlikView EE Test Server and QlikView SBE Test Server, both of which have the same features and limitations as the corresponding production servers. In addition, the watermark “Test” is superimposed on all charts and added to all object captions.<br>**Note!** There is no license lease from QlikView Test Server. |

| Product | Description |
|---|---|
| QlikView Publisher | QlikView Publisher is a license that adds significant functionality to the standard reload capability of QlikView Server. QlikView Publisher includes functionality to handle field level security and access control from central administration software like Window Active Directory or Novell® LDAP. QlikView Publisher is also needed to support complex distribution models for QlikView documents. In addition, each license of QlikView Publisher allows an additional node/server for reload, distribution, or security management in a multi-node/server deployment.<br>With the additional component QlikView Publisher Report Distribution, any QlikView document report can be distributed as a `.pdf` file to a folder or via email or SMTP. |
| QlikView WorkBench | QlikView WorkBench (an add-on to QlikView EE Server) is a development tool for creating web mash-ups with QlikView. It features drag and drop editing capabilities within the Microsoft Visual Studio® development environment and allows for custom web interfaces and integration with third-party services.<br>**Note!** QlikView WorkBench is not available for use with QlikView SBE Server. |
| QlikView Web Parts for Microsoft SharePoint® | QlikView Web Parts (an add-on to QlikView EE Server) for Microsoft SharePoint allows for rapid deployment of QlikView objects within Microsoft SharePoint portal environments.<br>**Note!** QlikView Web Parts are not available for use with QlikView SBE Server. |
| QlikView Local Client | QlikView Local Client is a client with all functionality used to develop QlikView documents. QlikView Local Client is deployed, if the end user deploys local clients only. |
| QlikView Personal Edition | Anybody registered on QlikView.com is allowed to download QlikView and develop QlikView documents for personal use. There are no restrictions to QlikView Personal Edition except that it cannot open QlikView documents created by other users or perform an import of an entire layout from an XML file. |

## 15.2 Features and Limitations

The table below lists the features and limitations of each edition of QlikView Server (Yes = supported, No = not supported).

| | EE | SBE | IAS | QES |
|---|---|---|---|---|
| **Licensing** | | | | |
| Named User CALs | Yes | Yes (max 25) | No | No |
| Session CALs | Yes | No | Yes (unlimited) | Yes |
| Usage CALs | Yes | No | No | Yes |
| Document CALs | Yes | Yes (max 100) | No | No |
| External Users Allowed? | No | No | Yes | Yes |
| **Clients** | | | | |
| AJAX (and mobile devices via AJAX) | Yes | Yes | Yes (WorkBench included) | Yes (WorkBench included) |
| Internet Explorer Plugin | Yes | Yes | No | No |
| Installed QlikView Client | Yes | Yes | No | No |
| **Scalability** | | | | |
| Can be clustered (additional server license required) | Yes | No | Yes | Yes |
| Unlimited Documents | Yes | Yes | No (1 document only) | No (3 documents only) |
| **Integration** | | | | |
| Third Party Security Integration | Yes | No | No | Yes |
| Dynamic Data Update | Yes | Yes | Yes | Yes |
| **Features** | | | | |
| License Lease (offline access, Named User CALs required) | Yes | Yes | No | No |
| Annotations | Yes | Yes | No | Yes |
| Collaboration (sheets, sheet objects, and input fields) | Yes | Yes | No | No |
| Session Collaboration | Yes | Yes | No | Yes |

| | EE | SBE | IAS | QES |
|---|---|---|---|---|
| QlikView Publisher and PDF generation (additional license required) | Yes | Yes | No | No |
| QlikView Connector for use with SAP NetWeaver® (additional license required) | Yes | Yes | Yes | Yes |
| Test Server Option | Yes | Yes | Yes | Yes |
| Can be embedded in Microsoft SharePoint (QlikView Web Parts for Microsoft SharePoint) (additional license required) | Yes | No | Yes | Yes |
| Build bespoke mashups/AJAX applications (QlikView WorkBench) (additional license required) | Yes | No | Yes (included) | Yes (included) |
| **Security** | | | | |
| Section Access | Yes | Yes | No | Yes |
| Document Metadata Service (DMS) | Yes | No | No | Yes |
| Active Directory/NTFS | Yes | Yes | No | Yes |
| Anonymous User | Yes (with Session CALs) | No | Yes (mandatory) | No |

# Part 6 Appendix

# 16 Silent Installation

When running a silent installation, QlikView is installed with a limited set of or no dialogs at all. This means all features, properties, and user selections have to be known when creating the silent installation package. There are also some standard properties in Windows Installer Service that may be required.

To prepare a silent installation, the MSI file has to be extracted from the QlikView `Setup.exe` file.

A silent installation can be run with different interface levels:

`/qn` Completely silent.

`/qb` Basic user interface.

Add a + sign at end of the interface levels command to get a modal dialog at the end of the installation saying "Finished" and if it was successful or not.

The following silent installation command lines are recommended for QlikView:

```
msiexec /i QlikViewServerx64.msi Addlocal="all" IS_NET_API_LOGON_
USERNAME="Domain\username" IS_NET_API_LOGON_PASSWORD="password /qn+
```

Alternatively:

```
QlikViewServer_x64Setup.exe /s /v"/qn+ Addlocal="all" IS_NET_API_LOGON_
USERNAME="Domain\username" IS_NET_API_LOGON_PASSWORD="password"
```

The command line above installs all features completely silently with a modal dialog at the end of the installation.

If just a limited set of the features are to be installed, change `all` to the name of the feature instead. If several features are to be installed, separate them with commas.

The following features can be installed:

- DirectoryServiceConnector
- ManagementService
- QVS
- QvsDocs
- WebServer
- DistributionService
- SupportTools
- QvsClients with the sub-features Plugin and AjaxZfc
- MsIIS with the sub-features QvTunnel and QlikView Settings Service

**Note!** For the sub-features to be included in the installation, they have to be included in the list of features to be installed.

```
msiexec /i QlikViewServerx86.msi ADDLOCAL="all" DEFAULTWEBSITE="2" /qn+
```

This command line installs all features, including the virtual directories to another website than the default one. This requires a machine with Microsoft Internet Information Services (IIS) installed and more than one website on it. The site number also has to be known. Set `DEFAULTWEBSITE` to the site number where the virtual directories are to be installed. To find the number of the website, check IIS.

The installation procedure can be logged, using the following command:

```
msiexec /i QlikViewServerx86.msi ADDLOCAL="all" DEFAULTWEBSITE="2"/L*v
log.txt /qn+
```

# 16.1 Settings

The following settings are good to know when designing a silent installation package:

| | |
|---|---|
| **Prerequisites** | .NET Framework 4.0 |
| **Default installation folder (INSTALLDIR)** | ProgramFilesFolder\QlikView |
| **Windows Installer Version** | 3.1 Schema 301 |
| **Default language** | English (United States) 1033 |
| **Require Administrative Privileges** | Yes |
| **INSTALLEVEL** | 100, all features is set to 101 by default |
| **Features** | See *Silent Installation (page 93)*. There is a hidden feature called "Install". Do not remove it. |
| **IIS** | Four virtual directories and an Application pool are installed |
| **Services** | Five services are installed |

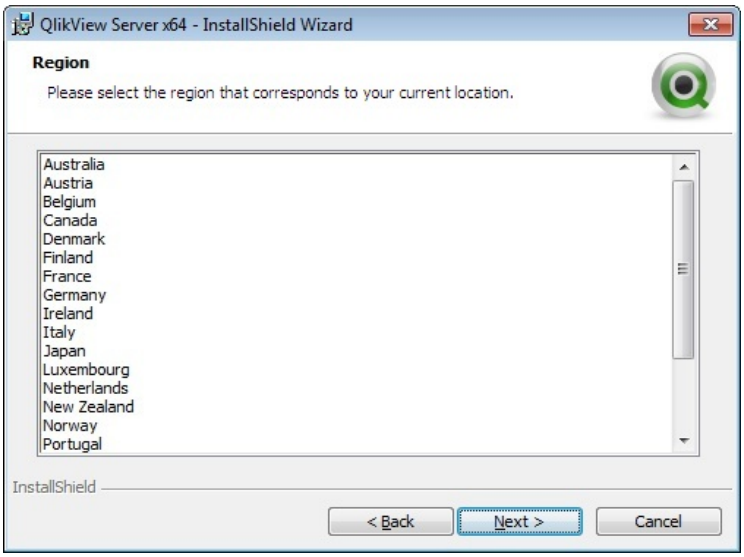
# 16.2 Dialogs

The QlikView installation has a number of dialogs, one of which is a Custom Setup dialog and one of which is a Website dialog. All dialogs set important properties. To find the value of a property, do a test installation with verbose logging. Note that the property values may differ depending on the language and operating system used.

## Region

This dialog is used for specifying the region.

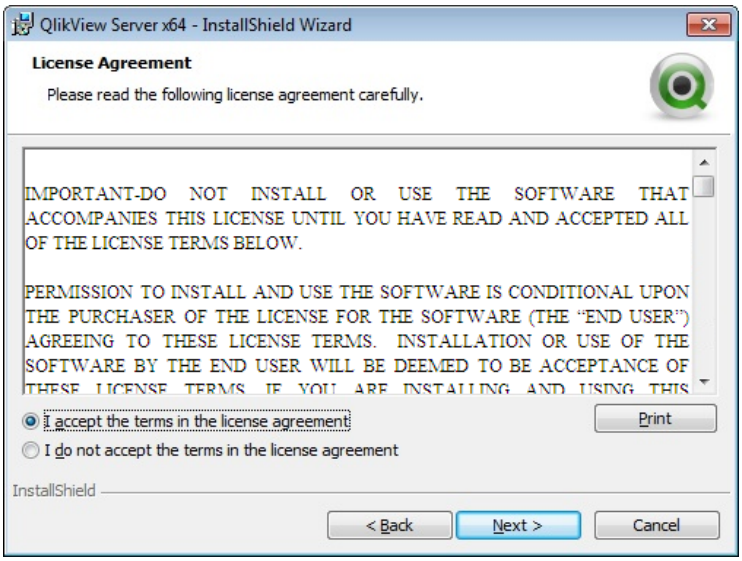
Property: `REGION_LIST`




*Region dialog*

# License Agreement

This dialog displays the license agreement for the selected region.
Radio button: `AgreeToLicense = "Yes"`

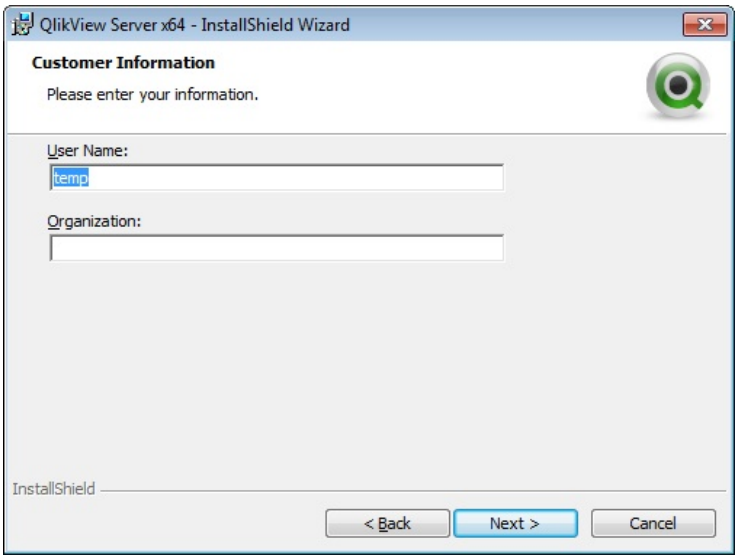



*License dialog*

# Customer Information

This dialog is used for entering the customer information.
Properties:

- `USERNAME`
- `COMPANYNAME`

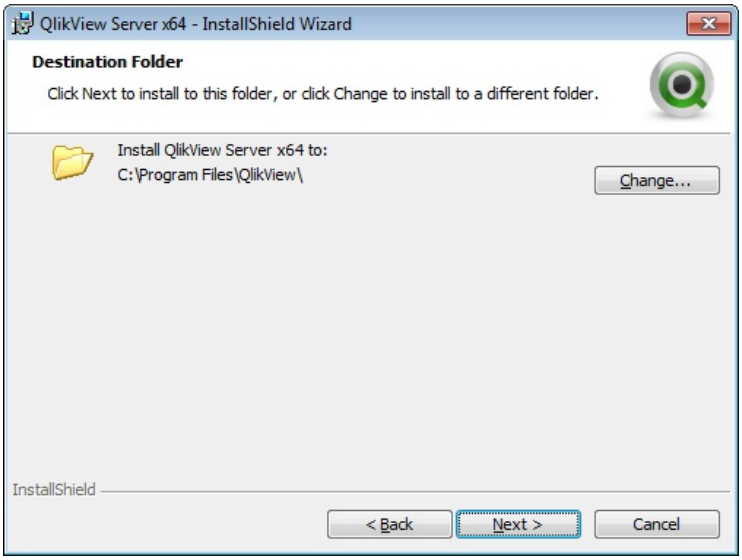



*Customer information dialog*

# Destination Folder

This dialog is used to set the default folder for the installation.

Property: `INSTALLDIR`




*Destination folder dialog*

# Profiles

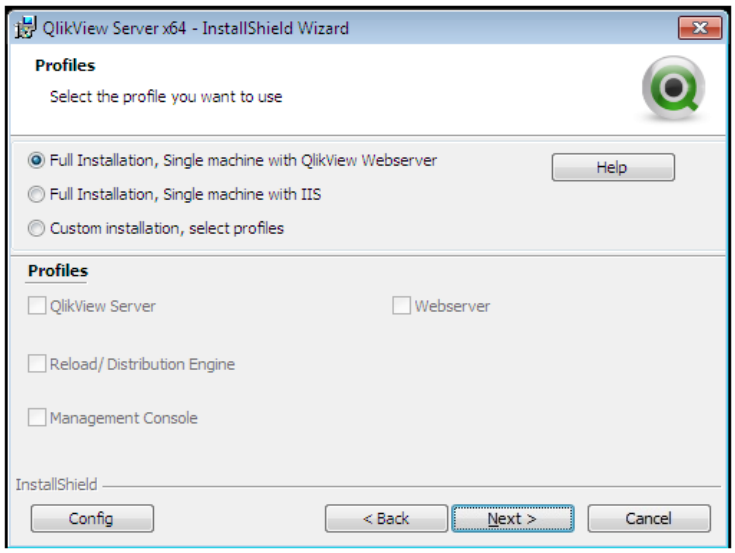
This dialog has several properties connected to it, since there are multiple profiles to choose from.

Select **Full Installation, Single machine with QlikView Webserver** to install everything, including QlikView Web Server, needed to run QlikView on a single machine. To use IIS instead, select **Full Installation, Single machine with IIS** (this option is only available if IIS is installed on the target machine).

To perform a custom installation, select **Custom installation, select profiles** and then select the profiles to install. The **Webserver** profile allows the user to choose between QlikView Web Server and IIS (if IIS is installed on the target machine).

Properties:

- `PROPQVS`: QlikView Server
- `PROPDS`: Publisher
- `PROPQMC`: Management Console
- `PROPWEB`, `PROPIIS` = 1 or 2: Webserver
- `PROPIIS` (if IIS is installed) or `PROPSTATE`: Single Machine Install



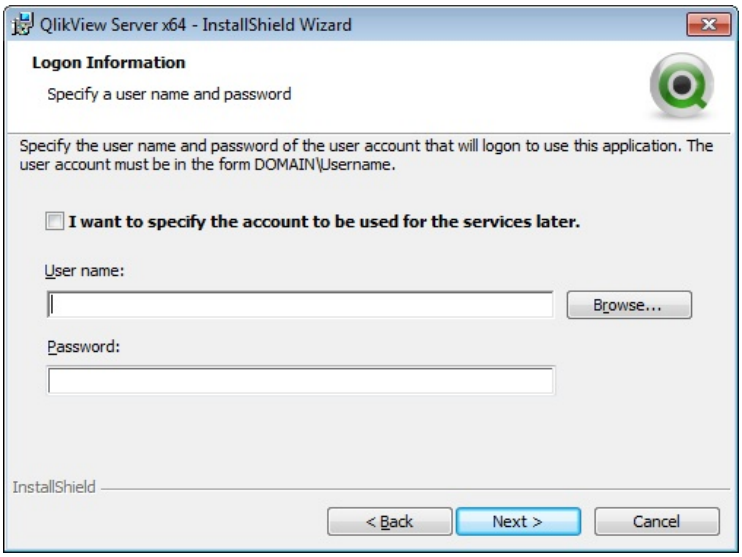

*Profiles dialog*

# Logon Information

This dialog, which is optional to use, is used to specify the user that is to run the services that are installed. When clicking **Next**, a Custom Action checks that the entered user is valid. The Custom Action, which is implemented by InstallShield, requires the machine to be part of a Domain to work properly.

Properties:

- `LOCALSERVICE`
- `IS_NET_API_LOGON_USERNAME`
- `IS_NET_API_LOGON_PASSWORD`

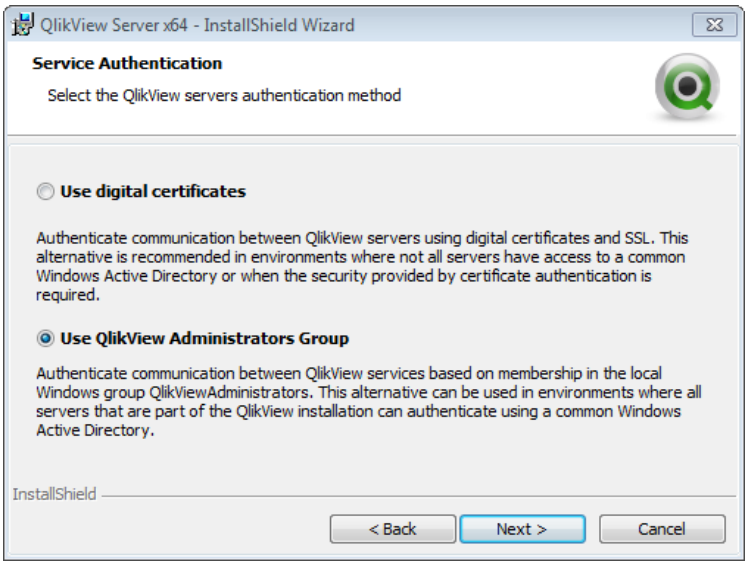



*Logon information dialog*

# Service Authentication

This dialog is used to select the type of service authentication. QlikView Administrators Group is selected by default.

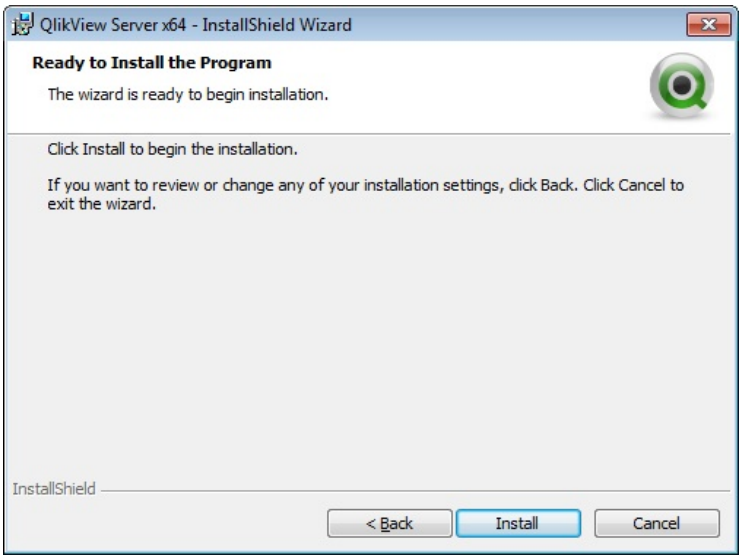
Property: `PROPCERT` (1 = Digital certificates, 2 = QlikView Administrators Group)




*Service authentication dialog*

## Ready to Install

This is the last dialog. Click **Install** to start the installation.

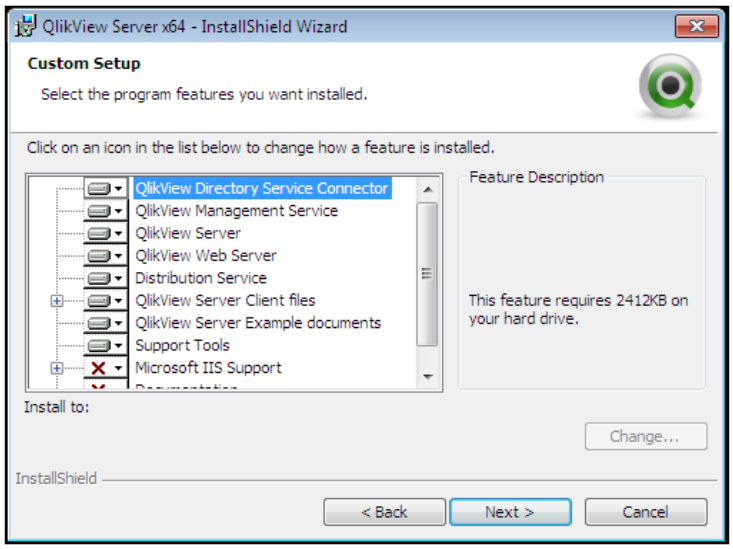



*Ready to install dialog*

# 16.3 Additional Dialogs

## Custom Setup

This dialog is displayed when clicking **Config** in the Profiles dialog, see *Profiles (page 96)*.

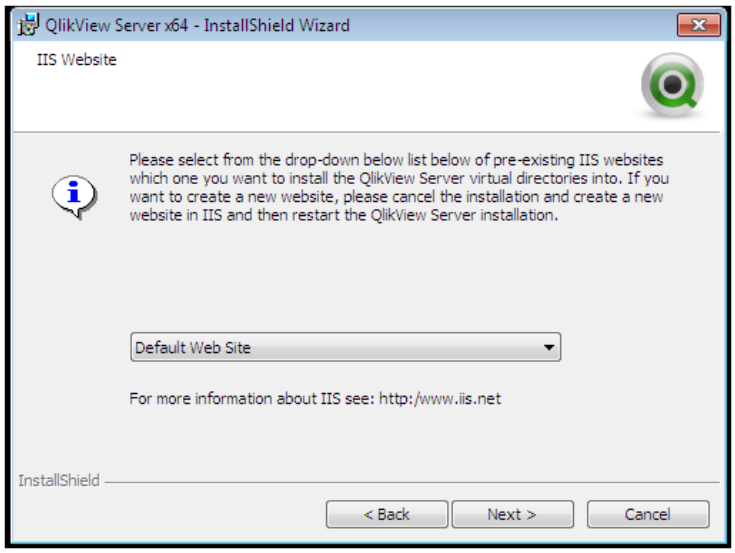



*Custom setup dialog*

## Website

This dialog is displayed when selecting IIS as web server in the Profiles dialog, see *Profiles (page 96)*.

Property: `DEFAULTWEBSITE`




*Website dialog*

# 16.4 MST

When creating an MST file, the MSI file is customized without any changes being made directly in the MSI. The MST file works as a filter on top of the MSI and allows changes to be made to the installation. For example, the default installation folder for QlikView Server is `%ProgramFiles%\QlikView`, but if that is changed to `C:\QlikView` in the MST file, the default folder is changed. The same thing can be done with the dialogs, which means properties can be preset, so that the installation can be run with a limited set of dialogs.

To create an MST file, an MSI repackaging studio (for example, InstallShield AdminStudio) is needed.

**Note!** QlikTech does not supply any MST files and does not take any responsibility for MST files created by customers or partners.

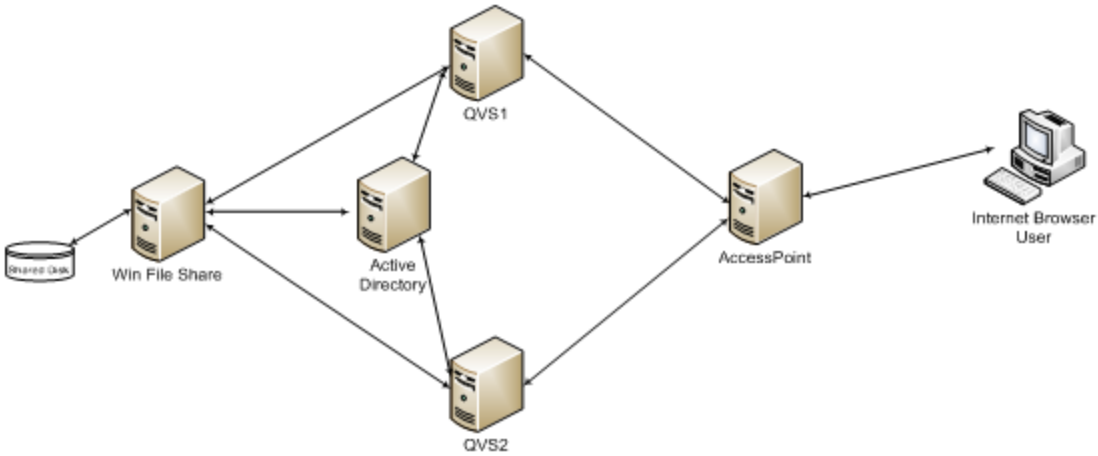
## 16.5 Additional Information

For additional information on silent installation, see *Deploying MSI Packages with Group Policies (page 129)*.

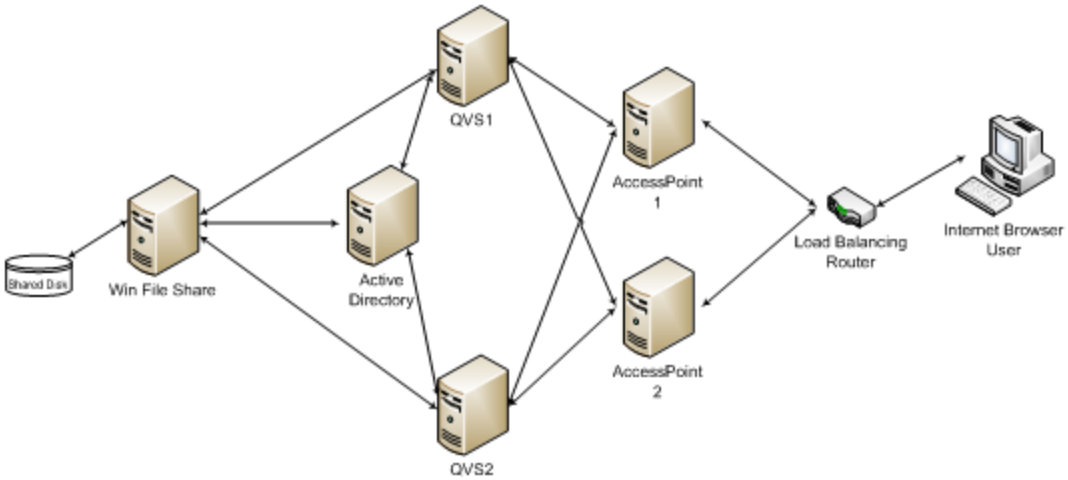
# 17 Clustering QlikView Servers

This chapter discusses the architectural and installation requirements and options for building a clustered and resilient QlikView Server deployment.

The following figure shows a clustered, load balanced QlikView Server deployment that uses AccessPoint (software load balanced).




The following figure shows a resilient, clustered, load balanced QlikView Server deployment that uses AccessPoint and network load balancing.




The QlikView Server load balancing capabilities are included in the QlikView web portal, AccessPoint. This chapter also discusses how to make this component resilient using network load balancing (if needed).

## 17.1 Why Cluster QlikView Servers?

By clustering QlikView Servers, the objectives described below can be achieved.

### Horizontal User Scalability

If more resources than can be provided by a single QlikView Server are needed, an additional server can be added. For example, if the server can support 1,000 concurrent users, but 2,000 concurrent users have to be supported, an additional server can be added. In this scenario, the first 1,000 users could be allocated to

server A and the second 1,000 users to server B. Alternatively, the servers could be clustered so that, to the end users, there is just one "world" (in reality it would be a single IP address or URL).

## Resilience

When the number of users increase, so does the users' reliance on QlikView. By clustering the QlikView Servers, resilience can be built into the deployment. In the case above, where a single server can support 1,000 users, three servers could be used to build resilience into the deployment. This would allow one server to be lost (due to, for example, hardware failure) with the system still capable of supporting 2,000 users. Having all three servers as active nodes helps reducing the response times by not running all servers at 100% of their capacity and also limits the number of users affected if a node is lost.

However, QlikView currently does not provide any session recovery options. In practice, this means that if a node in the QlikView cluster is lost, the users lose the analysis they are currently performing and that they have to reconnect to the cluster to resume their work. This does not mean that the data within the QlikView application is lost and needs to be reloaded, as the data is stored in the `.qvw` file on the NAS.

# 17.2 Requirements for Clustered QlikView Deployment

There are four high-level requirements for building a clustered QlikView deployment:

1. Clustered QlikView Server license key
2. Shared storage area
3. AccessPoint load balancing strategies
4. Network load balancer for providing full resilience (optional)

## Clustered QlikView Server License Key

In a clustered environment, the QlikView Server machines are installed with the same license key, which must be enabled for clustering. This can be checked confirmed by examining the following entry in the License Enabler File (LEF):

`NUMBER_OF_CLUSTER_NODES; 2` (number of nodes in the cluster)

Clustered QlikView Servers share configuration and license information between themselves via the shared storage, so that configuration and license management only needs to be performed once from the QlikView Management Console (QMC) for all nodes.

The servers must be installed on the same network subnet and have a shared root document directory; hence the requirement for a shared network storage. The configuration information is stored in Persistent Global Objects (`.pgo`) files.

If the servers fail to start or reset after ten minutes, check for the LEF entry above. This is usually an indication of multiple non-clustered servers with the same license key being used.

## Shared Network Storage

Shared network storage is required not only for the `.pgo` files mentioned above, but also for storage of QlikView applications that are required in the cluster. This also enables collaborative objects to be shared across the nodes in the cluster (using `.shared` files).

This shared storage area is the "Shared Disk" located on the left hand side of the figures in *Clustering QlikView Servers (page 101)*. A clustered QlikView deployments uses a Windows Server-based system.

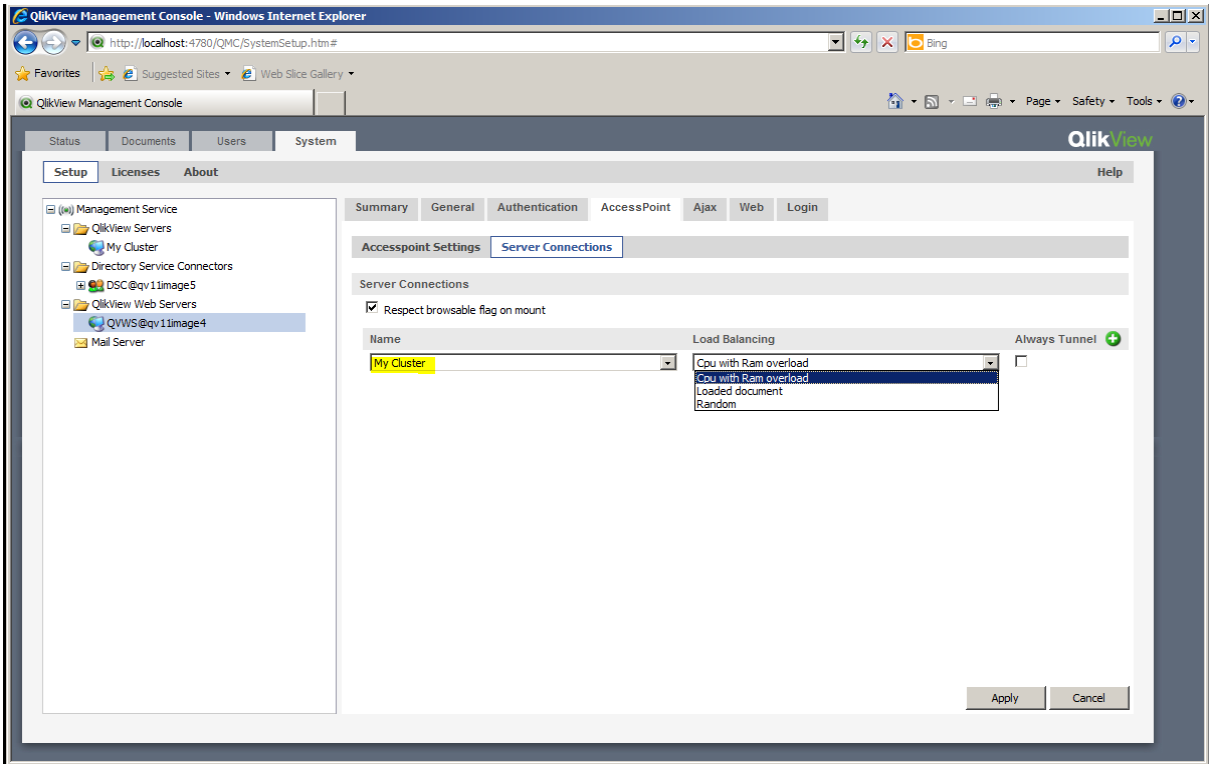
QlikView requires the storage of documents (`.qvw` files), `.pgo`, `.meta`, and `.shared` files to be hosted on a Windows-based file share. Hosting files on any other type of system is unsupported and may create an unstable QVS cluster where CALs disappear and QVSs stall. QlikView supports the use of a SAN (NetApp, EMC, etc) with a QlikView Server, provided it is mounted to a machine running Windows Server 2003 or later and then shared from that server.

# AccessPoint Load Balancing Strategies

QlikView AccessPoint supports three load balancing strategies:

- Random (default setting): A round robin type strategy ideal for most users, since the session is distributed across all nodes in the cluster.
- Loaded document: Used when sessions for the same document are to be routed to the same server. This strategy is designed for deployments where there are more documents than a single node in the cluster can handle. AccessPoint makes the decision based on if the document is already loaded and on the amount of RAM available on the server.
- CPU with RAM overload (only available in QlikView 11): Allows QlikView Web Server (QVWS) to route traffic based on two factors, (1) RAM and (2) CPU use. The node is chosen using the following criteria:
  - If RAM is readily available (low) on all available nodes, choose the node with the lowest CPU use.
  - If RAM is moderately used on all available nodes, choose the node with the most RAM available.

The QlikView load balancing strategy can be set in the QMC under **System>Setup>QlikView Web Servers**. Select the web server (either IIS or QVWS) on the **AccessPoint** tab:




# Network Load Balancer (Optional)

The network load balancer provides the resilience for AccessPoint, routing the sessions to an available AccessPoint server.

There are several requirements on the load balancer:

- Support for “sticky sessions”: This ensures a user's session persists on the same node within the cluster, usually by using a cookie.
- Availability: The load balancer checks the availability of the AccessPoint web server and the QlikView servers.
- Some form of load balancing algorithm to determine which server is the least loaded.

### Sticky Sessions

The requirement is for the user's session to be routed consistently to the same server. Methods for doing this vary from device to device – refer to the load balancer documentation for information on the options available.

### Availability Checking

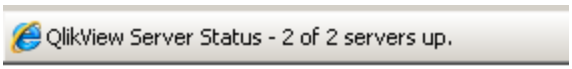
A special web page on the AccessPoint provides automated checking of the system status:

**http://myAccessPoint/QvAjaxZfc/QvsStatus.aspx**

This page returns an http status code of 200, if the AccessPoint and at least one QlikView Server in the cluster respond. Any other status code returned by this page should be considered an error. Common errors from this page include:

- 404: The AccessPoint is unable to respond. Check the web server.
- 503: No QlikView Servers responded to the AccessPoint and therefore it cannot service user requests.

The status of the QlikView Server cluster is also displayed on the web page:




| Name | Host | Status | Reason |
|---|---|---|---|
| MyCluster | qvs1 | Up | |
| MyCluster | qvs2 | Up | |

### Load Balancing Strategies

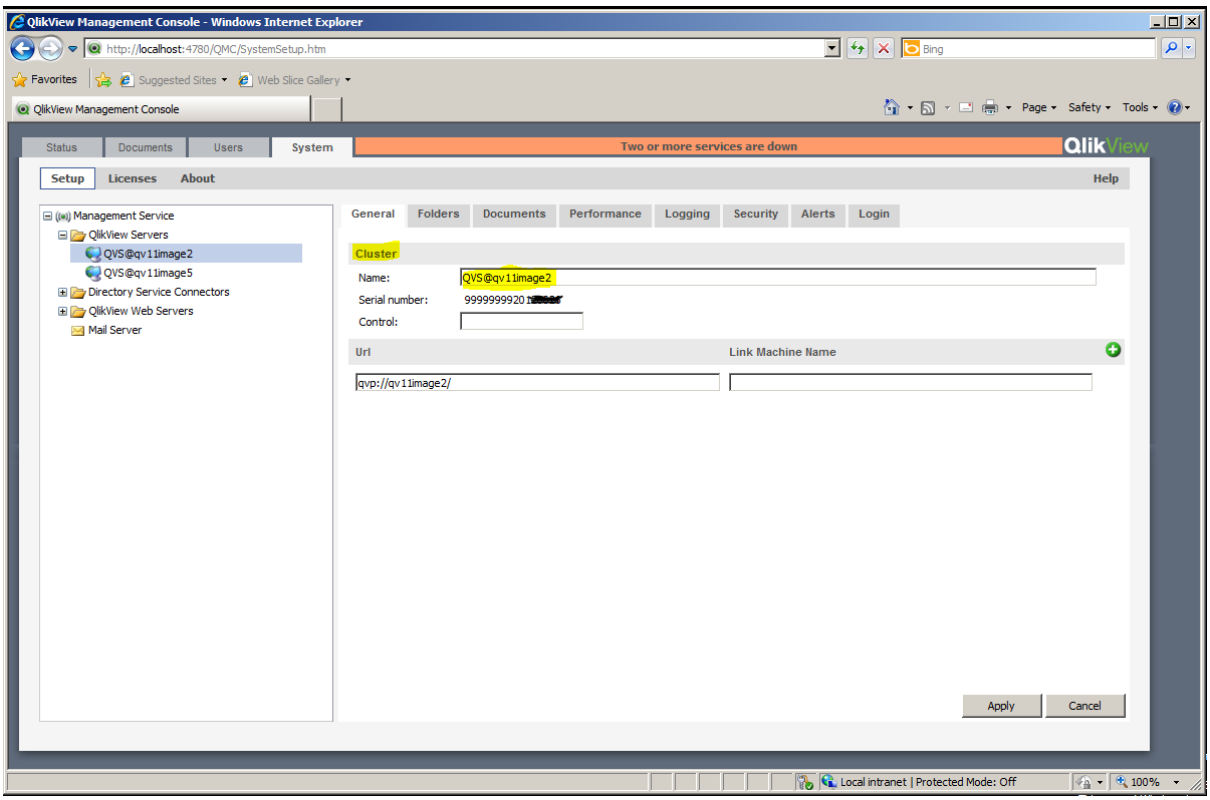
There are several strategies for how the load balancing router allocates sessions to the nodes within the cluster:

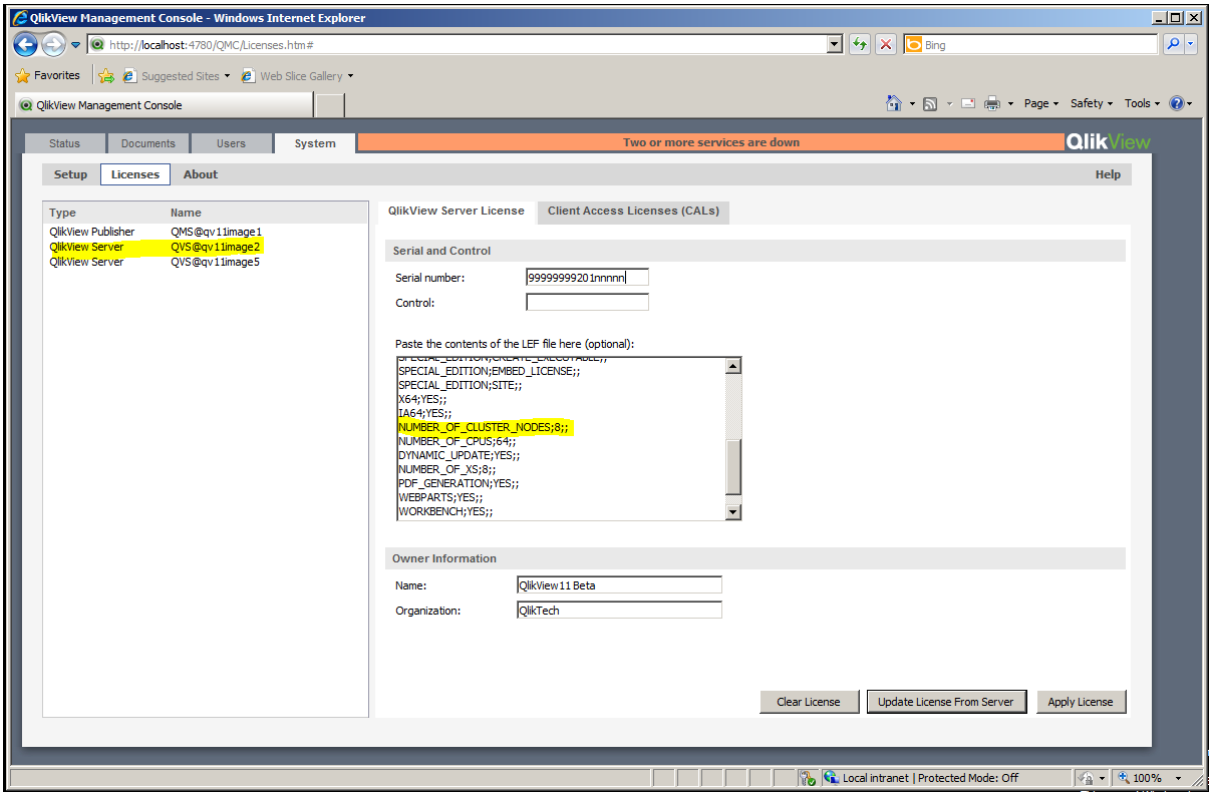
- Round robin: The load balancer sends each session to the next available server. This is a fairly rudimentary load balancing algorithm.
- Session counts: The load balancer keeps a running count of the number of sessions on each AccessPoint and ensures that there is an equal number of sessions on each node.

## 17.3 Building and Installing a QlikView Cluster

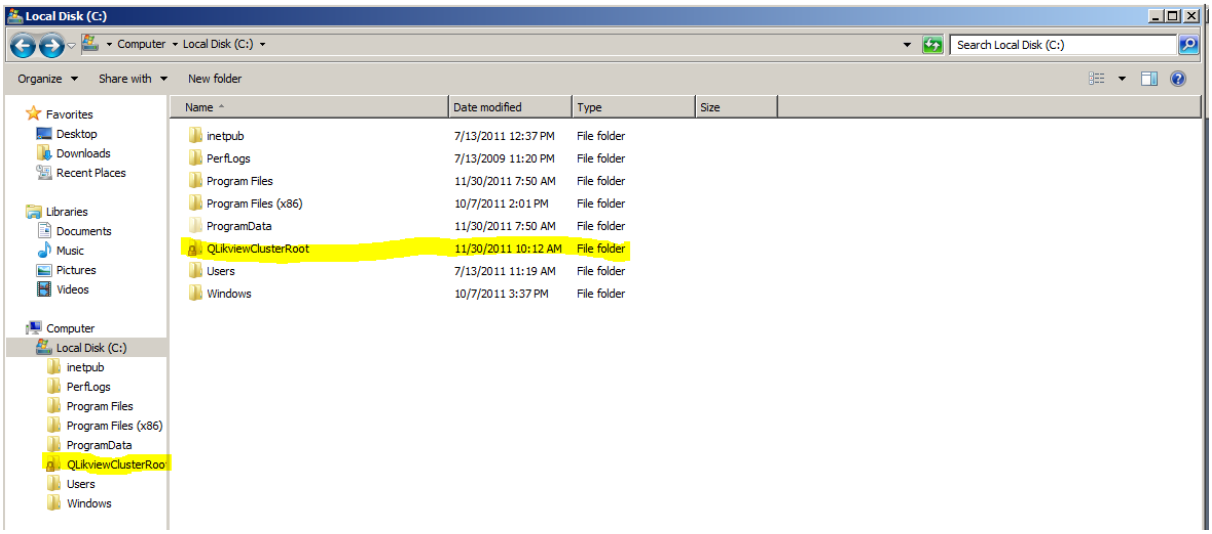
Proceed as follows to configure and activate a QlikView Server cluster using the QMC:

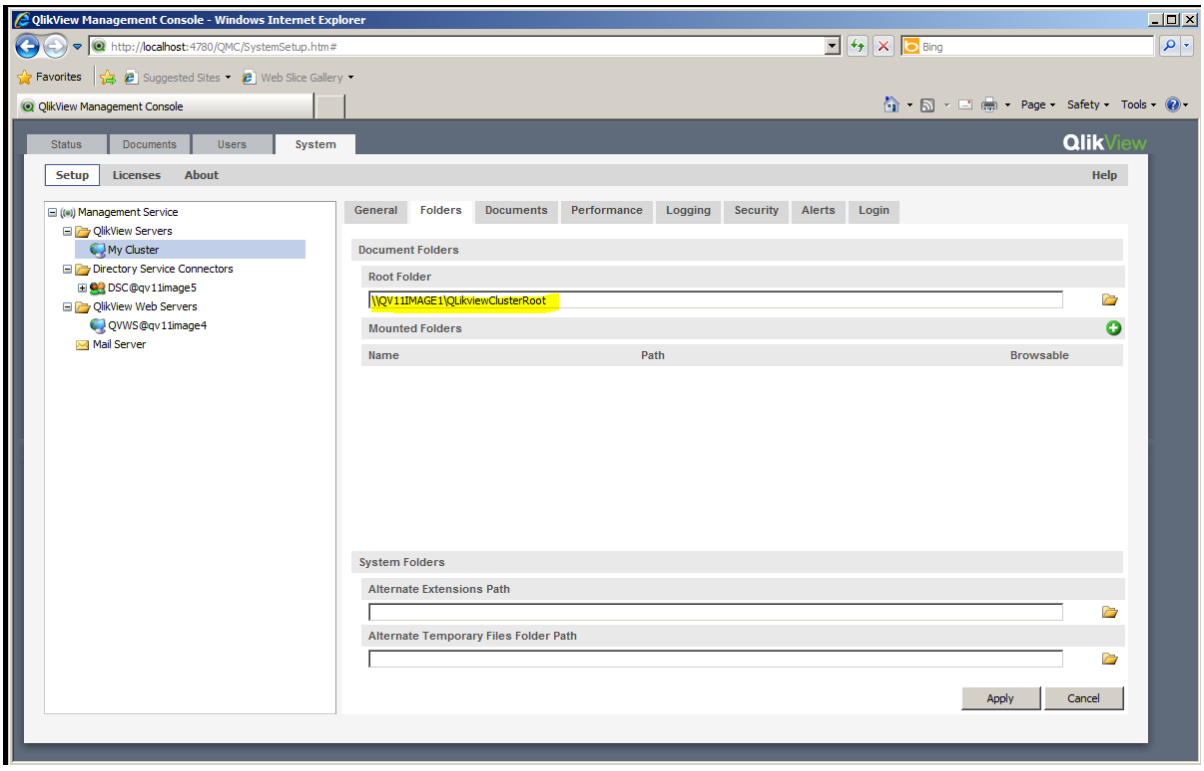
1. Install and license the first QlikView Server in the cluster. This will be the "master" QlikView Server.



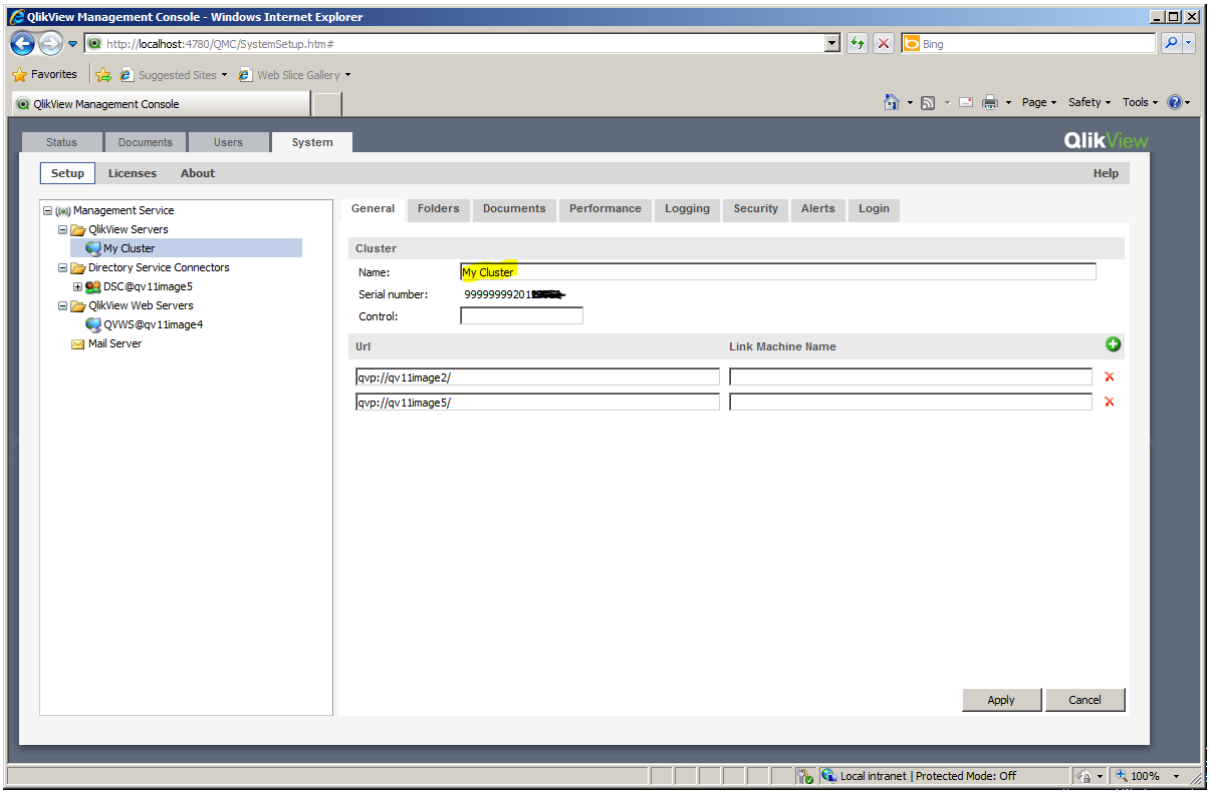




2. Configure the document folder to point to a folder on the NAS that all QlikView Servers in the cluster can access.

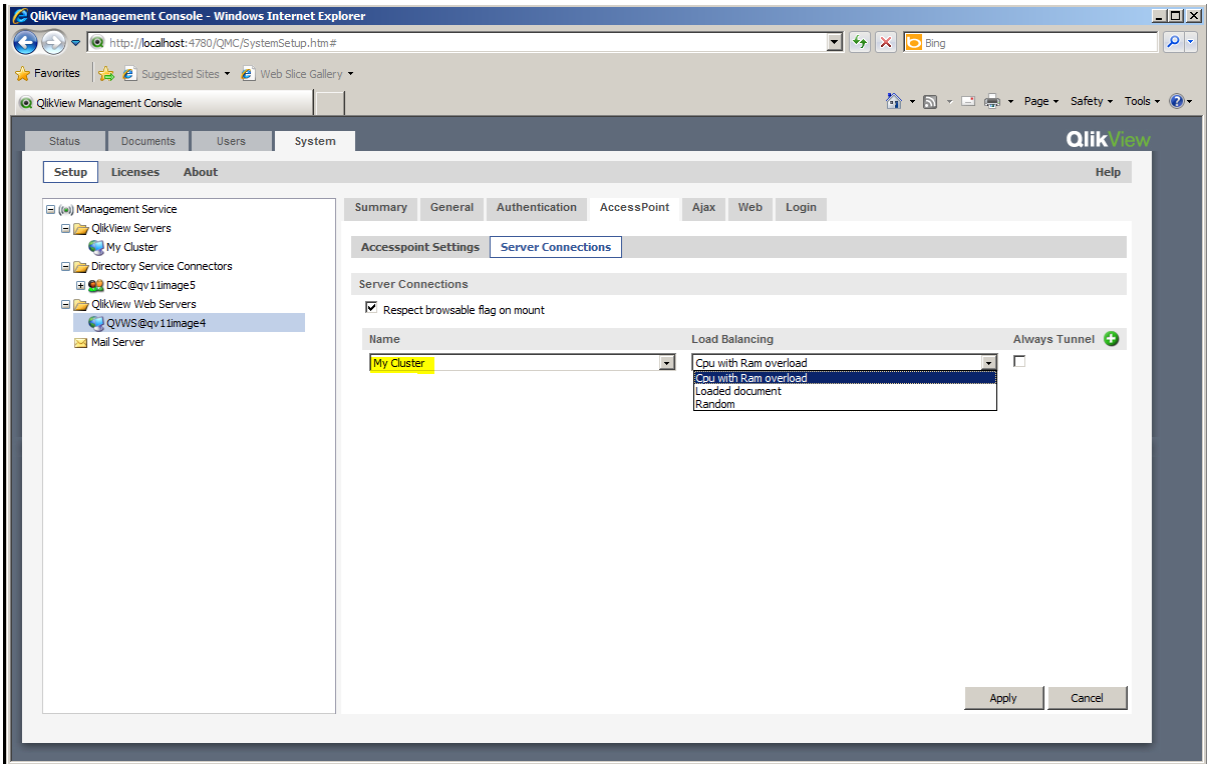



3. Install the next QlikView Server in the cluster.
4. Ensure that all QlikView services are running as local administrators and that they are members of the "QlikView Administrators" local group.
5. Open **System**>**Setup** in the QMC and select the server. Then go to the **General** tab and enter the control number for your license and the address to the second QlikView Server in the cluster.
6. If needed for usability reasons, go to the **General** tab for the QlikView Server in the QMC and rename the cluster (in this example, the cluster is renamed "My Cluster").
7. Repeat steps 3 - 5 for the QlikView Server nodes in the cluster.




8. Make sure that the cluster is selected in **Server Connections** in the settings for the AccessPoint.




9. The cluster is now configured and ready to use.

# 18 Clustering QlikView Publisher

This chapter provides an overview of QlikView Publisher and how to use it in a clustered deployment for scalability, resilience, or both. This chapter also addresses the architectural and installation requirements and the options for building a clustered and resilient QlikView Publisher deployment.

## 18.1 Introduction

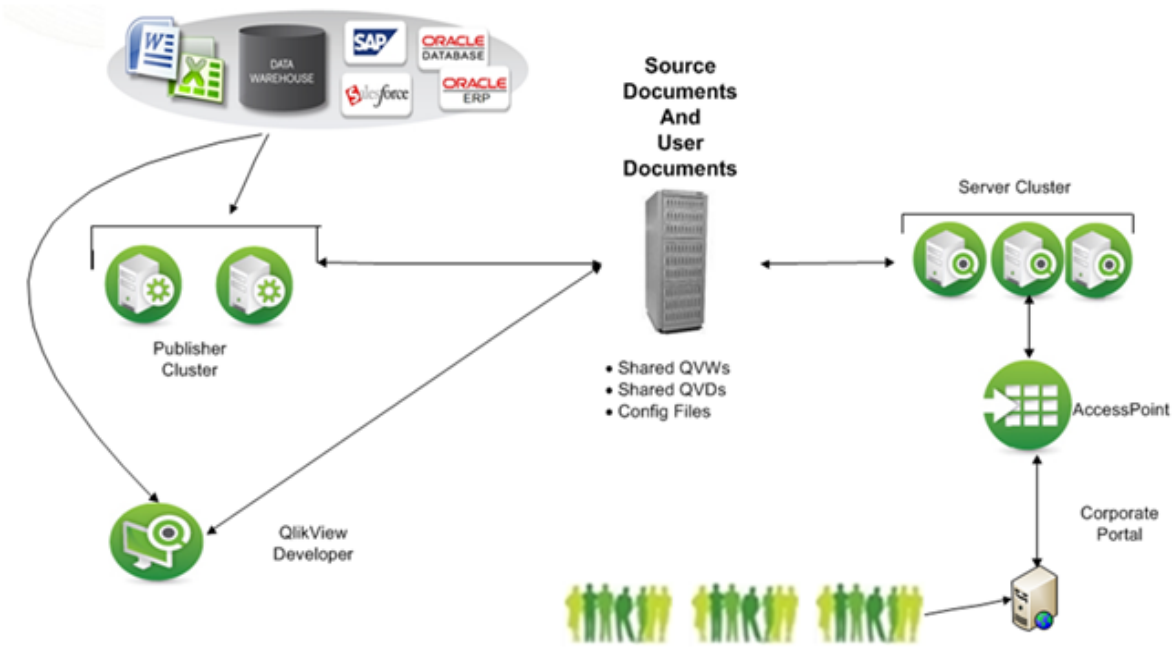
QlikView Publisher is an optional module for QlikView Server that enables scheduling, administration, and management tools that provide a single point of control for QlikView analytics applications and reports. Administrators can schedule, distribute, and manage security and access for QlikView applications and reports across the enterprise.

QlikView Publisher performs the following main functions:

- It loads data directly from data sources defined in connection strings in the source `.qvw` files.
- It is used as a distribution service to "reduce" data and applications from source `.qvw` files based on various rules (for example, user authorization or data access) and distribute these newly-created documents to the appropriate QlikView Servers or as static reports via email.
- When using QlikView Publisher, only Publisher has access to the source documents folder and the data sources for data load and distribution. The source documents and data are not accessible by QlikView users.

By deploying a clustered architecture, QlikView Publisher achieves scalability and/or resilience using web services technology. Administrators can cluster services together to provide load balancing. Native support for SNMP enables integration with enterprise system monitoring tools. External enterprise scheduling tools can trigger Publisher tasks using web service calls. Tasks can also be scheduled and executed on demand by QlikView administrators.

The figure below shows a two-server, clustered QlikView Publisher where each server is configured for processing different tasks and load balancing. The figure also includes a three-server, clustered QlikView Server that uses QlikView AccessPoint for load balancing. Documents created by QlikView Developer are stored in the source documents folder. QlikView Publisher tasks are used to retrieve data and store the result in the user documents folder.




## Source Documents

The source documents contain a) scripts within `.qvw` files to extract data from various data sources (for example, data warehouses, Microsoft Excel files, SAP, and Salesforce.com), b) the actual binary data extracts themselves within `.qvd` files, or c) a binary load from another `.qvw` file, inheriting its data model in one line of code.

The QlikView source documents, created using QlikView Developer, reside in the following folder:

- Windows Server 2008 and later: `\ProgramData\QlikTech\SourceDocuments`. This is the default QlikView location for Windows Server 2008 and later.
- Windows Server 2003: `\Documents and Settings\All Users\Application Data\QlikTech\SourceDocuments`. This is the default QlikView location for Windows Server 2003. However, for a QlikView Publisher cluster, this folder has to be relocated to a shared folder designated in the QMC Publisher configuration.

## User Documents

The user documents folder is the repository used by QlikView Server. The folder is located at:

- Windows Server 2008 and later: `\ProgramData\QlikTech\Documents`. This is the default QlikView location for Windows Server 2008 and later.
- Windows Server 2003: `\Documents and Settings\All Users\Application Data\QlikTech\Documents`. This is the default QlikView location for Windows Server 2003.

## Tasks

Tasks are created by administrators for data distribution and data reloads. Tasks are stored in the QlikView Publisher repository as a collection of XML files or in an SQL Server database. When a task is executed, QlikView Publisher invokes QlikView Batch (QVB), which is comparable to QlikView Desktop without the user interface.

**Note!** QlikView Batch (QVB) does not support graphical or user input objects. This means that QVB cannot reload documents that, for example, contain scripts that require user input.

QVB reloads the documents, which are stored in the source documents folder(s) and creates an associative QlikView database, which is stored within each document. The QVB performs the reload by retrieving the data described by the load script from the data sources. QlikView Publisher distributes the documents to the user documents folder for QlikView Server using the encrypted QVP protocol, to a mail server, and/or a file folder. QlikView Publisher can use the Directory Service Connector (DSC) to determine where and to whom the documents are to be distributed.

# 18.2 Why Cluster QlikView Publisher?

The role of Publisher in the QlikView solution is to distribute and refresh data by criteria set by the QlikView administrator. To accomplish this, Publisher executes many tasks, either scheduled or on demand. A Publisher task is the smallest entity that can be distributed in a cluster; a single task cannot be divided and executed in parallel on multiple cluster nodes. Clustering the Publisher service on more than one server enables the administrator to distribute multiple tasks to multiple servers operating in parallel using the Publisher load balancing algorithm. This means Publisher clusters can be used to increase the scalability, availability, and serviceability of data distribution and reloading.

In addition, a Publisher cluster license enables the configuration of Publisher services in clusters and standalone Publisher services. For example, a Publisher cluster can be used in a corporate office to handle large volumes of data and tasks, whereas a single Publisher service can be used in an associated manufacturing plant where the Publisher only needs to distribute documents using the manufacturing data source.

By clustering QlikView Publisher, the following objectives can be met:

- Horizontal scalability
- Resilience

## Horizontal Scalability

Horizontal scaling of hardware provides the ability to increase the resources of the QlikView deployment. By adding additional hardware servers, the workload of QlikView Publisher can be increased. The clustered Publisher servers can then be configured to load balance the QlikView tasks.

For example, on a certain hardware server, QlikView Publisher can process eight concurrent tasks. When the resource needs increase, the QlikView Publisher service can grow as needed. By adding an additional QlikView Publisher service on a new hardware server, the deployment can handle up to sixteen concurrent tasks by configuring the additional server in a Publisher cluster deployment. In this scenario, the first eight tasks are allocated to Server A and the second eight tasks to Server B. Alternatively, if the servers are clustered, the tasks can be load balanced over the two servers.

## Resilience

When the number of tasks in the deployment increases, the window for completing the tasks in time becomes increasingly important. Clustering the QlikView distribution services provides for resilience in the deployment. In the case above, where a single server can support 100 concurrent tasks, an additional server can be deployed (for a total of three servers) in order to build resilience into the deployment. If a server is lost (for example, due to a hardware failure or network connection issues), the resilient cluster still supports up to 200 tasks. Having all three servers as active nodes helps reduce response times by not running all servers at 100% of their capacity. It also limits the number of tasks and task chains affected if a node is lost.

# 18.3 Requirements for a Clustered QlikView Publisher Deployment

The following high-level requirements must be fulfilled for a clustered QlikView Publisher deployment:

- Clustered QlikView Publisher license key
- Shared network storage
- Load balancing strategies

# Clustered QlikView Publisher License Key

In a clustered environment, the QlikView Publisher servers are installed with the same license key. This can be verified by examining the following entry in the License Enabler File (LEF):

`PRODUCTLEVEL;30;;` (where `30` is the code for QlikView Publisher)

`NUMBER_OF_XS;N;;` (where `N` is the number of allowed QlikView Distribution Services)

The servers in a clustered QlikView Publisher deployment share configuration and license information among themselves via the shared storage, so configuration and license management only needs to be performed once in the QMC for all nodes.

# Shared Network Storage

Shared network storage is required for storage of QlikView applications that are needed in the cluster. It is recommended to host the storage of documents (`.qvw` files) and `.meta` data on a Windows-based file share. QlikView Publisher supports a SAN (NetApp, EMC, etc.) that is mounted to a Windows Server 2003 (or later) and then shared from that server. Storage presented to a server via a SAN must appear as locally attached storage. If SAN storage is used for Publisher, any distributed data that is accessed by QlikView Server should not reside on the SAN storage.

The QlikView Distribution Services (QDSs) must have a shared application data directory and possibly a shared source document directory as well (hence the requirement for a shared network storage). All configured Publisher services must have reliable network access to the shared storage.

# Load Balancing Strategies

## Load Balancing

The load balancing is determined by an internal ranking system based on the amount of memory available and the CPU use. QlikTech recommends using the default settings, since they have been extensively tested.

To change the default settings, edit the configuation file, `QlikViewDistributionService.exe.config`. The key is written in JavaScript:

```
<add key="LoadBalancingFormule" value="(AverageCPULoad*400) +
((MemoryUsage / TotalMemory) * 300) + ((NumberOfQlikViewEngines /
MaxQlikViewEngines)*200) + (NumberOfRunningTasks*100)"/>
```

where:

- `AverageCPULoad`: Average CPU load for all running QVBs.
- `MemoryUsage`: Total memory use for the entire application.
- `TotalMemory`: Total amount of memory on the server.
- `NumberOfQlikViewEngines`: Number of QlikView engines currently used.
- `MaxQlikViewEngines`: Configured value for the maximum number of QlikView engines.
- `NumberOfRunningTasks`: Number of tasks currently running.

## Simultaneous Tasks

By default, four QlikView tasks can execute simultaneously on a node. The recommended maximum is eight simultaneous tasks per node. If more than ten tasks have to be executed simultaneously on a node, modifications are necessary in the Windows registry to change the desktop heap size to allow for more simultaneous tasks.

**Note!** A large-scale server is required for executing ten or more simultaneous tasks. Alternatively, add additional servers for Publisher tasks.

Proceed as follows to change the number of tasks allowed to execute simultaneously:

1. Backup the Windows Server registry.
2. Locate the following Windows Server registry setting:

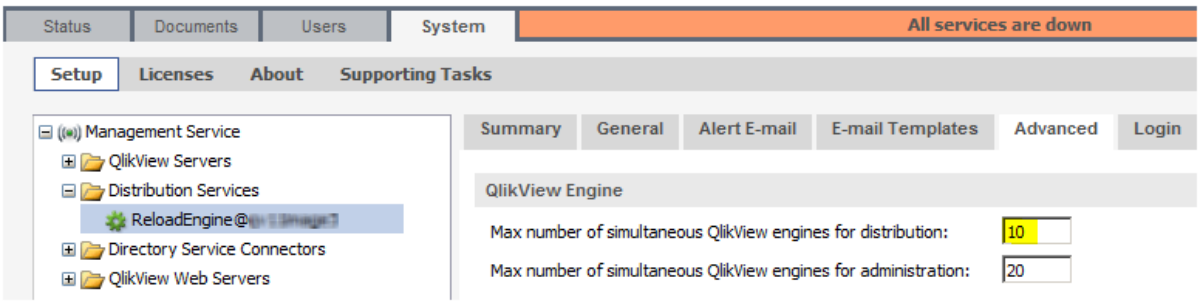
```
HKEY_LOCAL_
MACHINE\System\CurrentControlSet\Control\Session\Manager\SubSystems\Windows
%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
SharedSection=1024,3072,512 Windows=On SubSystemType=Windows
ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off
MaxRequestThreads=16
```

The default value for `SharedSection` is 1024,3072,512 for 32-bit (x86) and 1024,3072,768 for 64-bit (x64), respectively. For additional information, see **http://blogs.msdn.com/ntdebugging/archive/2007/07/05/desktop-heap-part-2.aspx**.

3. Change the "GDI" and "User handle max count" in the registry to `SharedSection=1024,20480,2048`:

```
HKEY_LOCAL_
MACHINE\System\CurrentControlSet\Control\Session\Manager\SubSystems\Windows
%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
SharedSection=1024,20480,2048 Windows=On SubSystemType=Windows
ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off
MaxRequestThreads=16
```

4. Change the **Max number of simultaneous QlikView engines for distribution** setting in QMC to the number of engines needed.

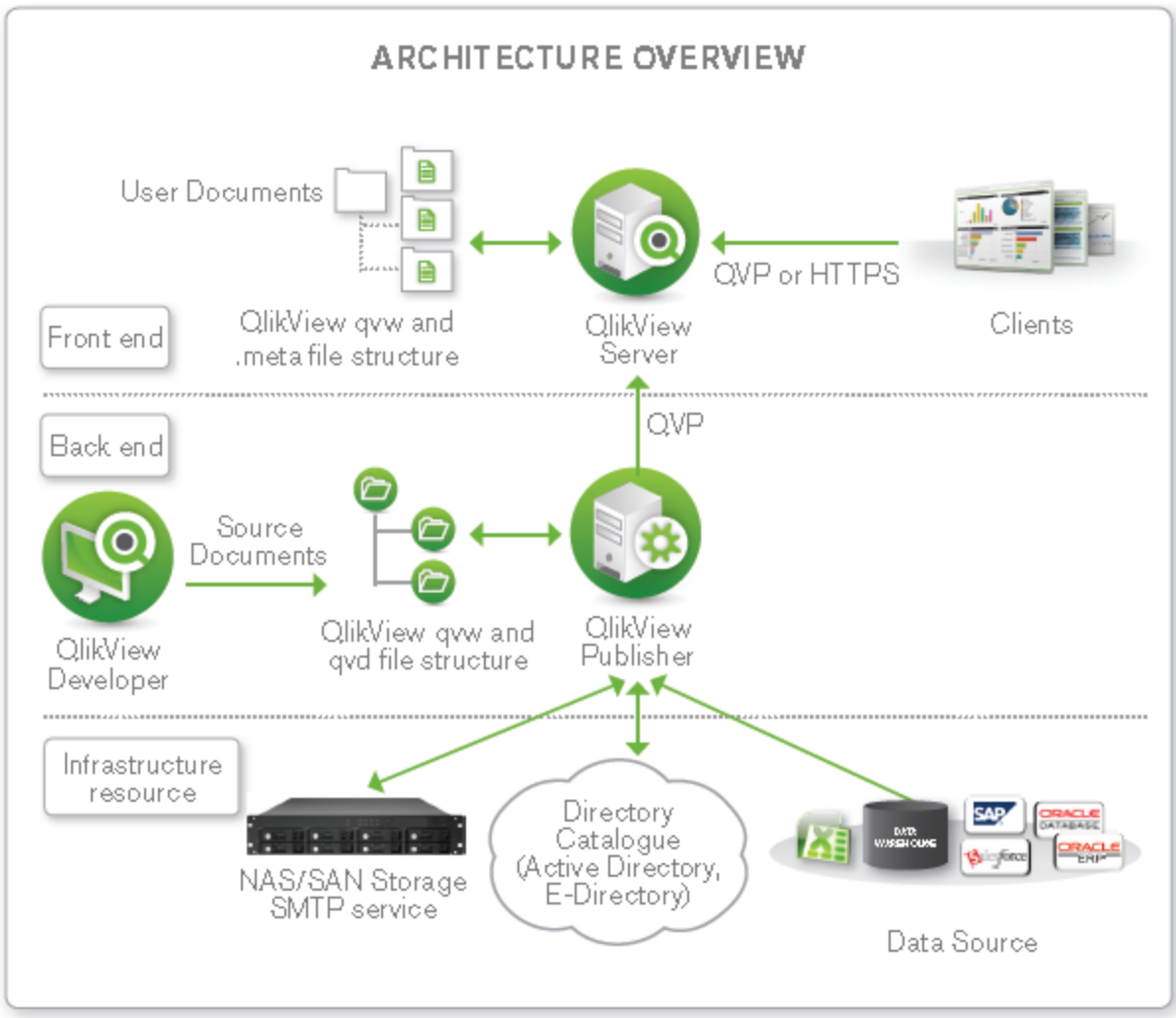



# 18.4 Security

QlikView Publisher provides access to QlikView applications and data. It is therefore important to integrate QlikView Publisher with the enterprise security solutions in addition to the standard security features of QlikView Server.

QlikView Publisher is viewed as a backend process within the QlikView solution. From a security perspective, it is important to understand that the frontend does not have any open ports to the backend. The frontend does not send any queries to data sources on the backend, nor do any of the user documents (`.qvw` files) contain any connection strings to data sources located on the backend. End users can only access QlikView documents that exist on the frontend. Within the backend, the Windows file system is always in charge of authorization; QlikView is not responsible for access privileges.

The figure below shows a simplified view of a standard QlikView deployment containing the location of the QlikView products and the data and applications.




## Directory Services

To provide security for QlikView documents, QlikView Publisher can connect to an external directory service (for example, Active Directory, LDAP, a database, or other sign-on solutions). The external directory service is an authentication source with which QlikView has a trust relationship.

QlikView provides a built-in Directory Service Provider (DSP) for Active Directory that allows QlikView administrators to assign Active Directory user privileges to QlikView documents or portions thereof. QlikView Publisher leverages this built-in provider to provide direct integration with, and support for, Active Directory.

QlikView also provides a means of creating a Configurable LDAP for other directory services. A Configurable LDAP enables QlikView administrators to grant privileges to users authenticated by any authentication system other than Active Directory.

## QlikView Server Authorization Modes

QlikView Server provides two mutually exclusive options for authorizing access to QlikView documents. Depending on the authorization mode of QlikView Server (NTFS or DMS), Publisher populates the appropriate Access Control List (ACL) when assigning rights to a document. In case of NTFS authorization,

Publisher populates a standard NTFS ACL when sending documents to QlikView Server. In case of DMS authorization, Publisher populates an ACL contained within a `.meta` file associated with the application.

## Static Data Reduction

Data reduction is a security mechanism that allows application data to be purged from a QlikView application in accordance with row-level security settings. QlikView Publisher can automate data reduction independently of the applicable security scenario. However, Publisher allows an administrator to configure data reduction based on users or groups defined within any external authentication source available through a custom or Active Directory DSP. Publisher performs the data reduction using the "loop and reduce" functionality in QlikView. The Publisher data reduction should not be confused with the dynamic data reduction associated with Section Access.

# 18.5 Configuring QlikView Publisher Clustering

**Note!** The instructions in this section are valid for Windows Server 2008 R2 and later.

## Requirements

The following requirements must be fulfilled before starting the QDS cluster configuration:

- A QlikView Publisher license that supports more than one QDS. The Publisher LEF must contain the entry `NUMBER_OF_XS;N;;`, where `N` is 2 or higher.
- QlikView AccessPoint (based on QlikView Web Server or Microsoft IIS), QlikView Management Service (QMS), QlikView Server (QVS), and DSC are already installed in the QlikView system in the network.
- A domain user to run the QlikView services on every machine is available.
- A shared storage device; QlikTech recommends a shared device mounted as a Windows-based file share.
  All QDS cluster nodes need read and write access to the following, centrally stored data:
  - QlikView Publisher status, configuration, and log files
  - QlikView source documents

## Step-by-step Instructions

### Prepare the Shared Storage Device

Create folders for the files accessed by every Publisher cluster node:

- `\\<server1>\ProgramData\QlikTech\DistributionService` (application folder)
- `\\<server1>\ProgramData\QlikTech\SourceDocuments` (source documents folder)

### Prepare the Cluster Nodes

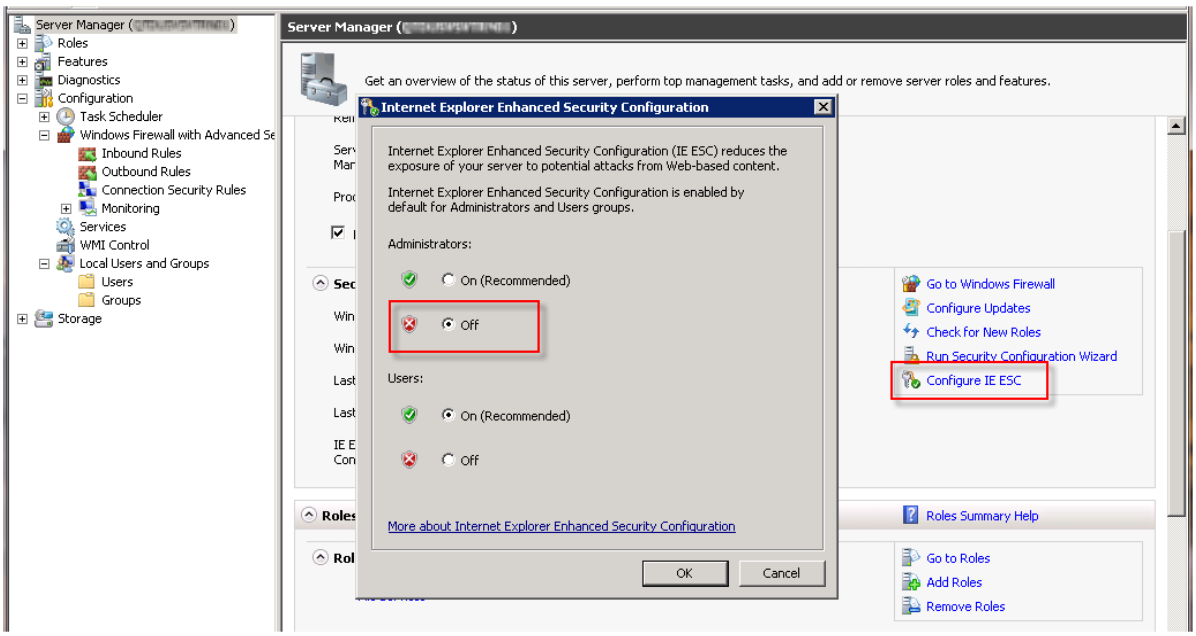
Proceed as follows on each planned QDS cluster node:

1. Login as administrator.
2. Configure the firewall to secure the QlikView solution. The QlikView services require the ports listed in the table below to be "opened".

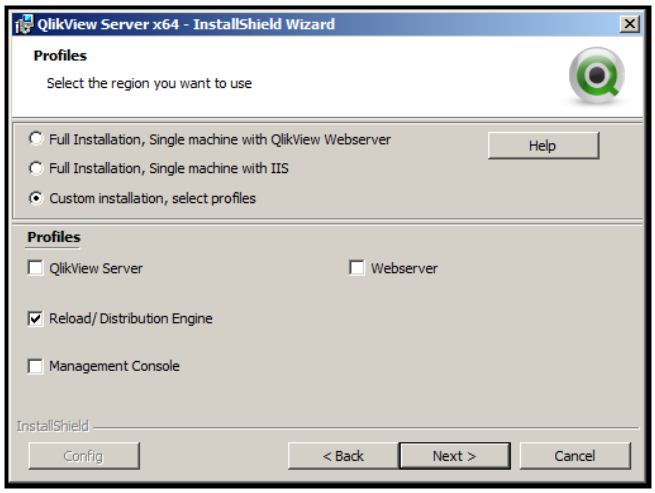
| Service | Port |
|---|---|
| QDS (Publisher) (required for Publisher) | 4720/TCP |
| DSC (required for Publisher) | 4730/TCP |
| QMS (required for Publisher) | 4780/TCP |

| Service | Port |
|---|---|
| QlikView Web Server/Microsoft IIS configuration | 4750/TCP |
| QVS configuration | 4749/TCP |
| QVP communication | 4747/TCP |
| QMS (EDX calls) (required for Publisher) | 4799/TCP |

3. Deactivate the Internet Explorer Enhanced Security Configuration for administrators. By default, Windows Server 2003 and later ship with this configuration enabled, which is basically a locked down version that adds a bit of extra security to the servers for web browsing. When the configuration is enabled, it may cause problems in viewing the QMC and service content. The Internet Explorer Enhanced Security Configuration can be left turned on, but if any issues arise, turn off the feature for the Administrators group.




4. Add the domain user that is used to run the QlikView services to the Local Administrators Group.
5. Start the QlikView 64-bit (x64) server setup and select **Custom installation, select profiles**. Then select the **Reload/Distribution Engine** feature and install it on each node where Publisher is to reside.

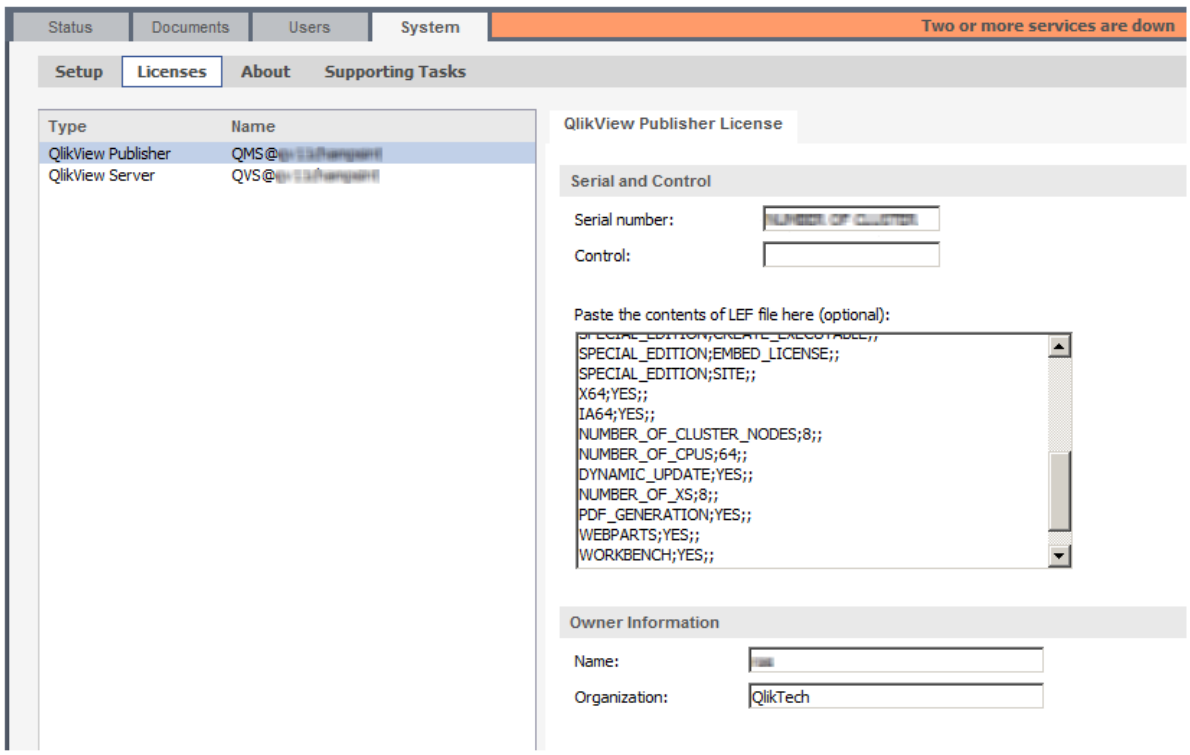



6. Enter the QlikView service account credentials.
7. Finish the setup and restart the system immediately.

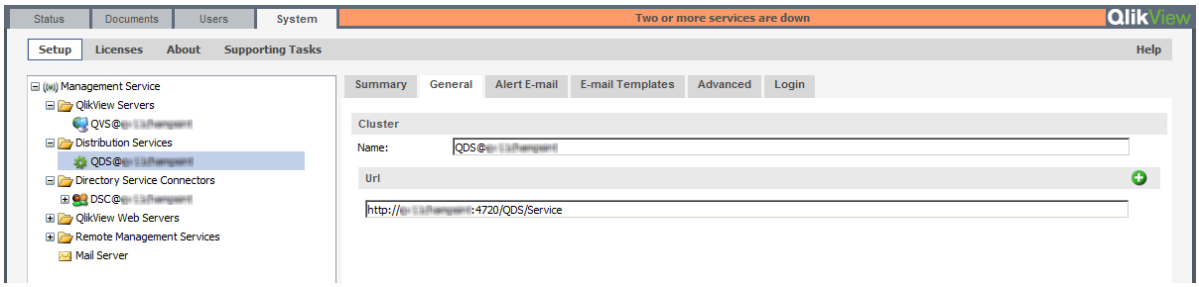
## Configuring QDS Cluster in the QMC

Proceed as follows to configure a QDS cluster in the QMC:

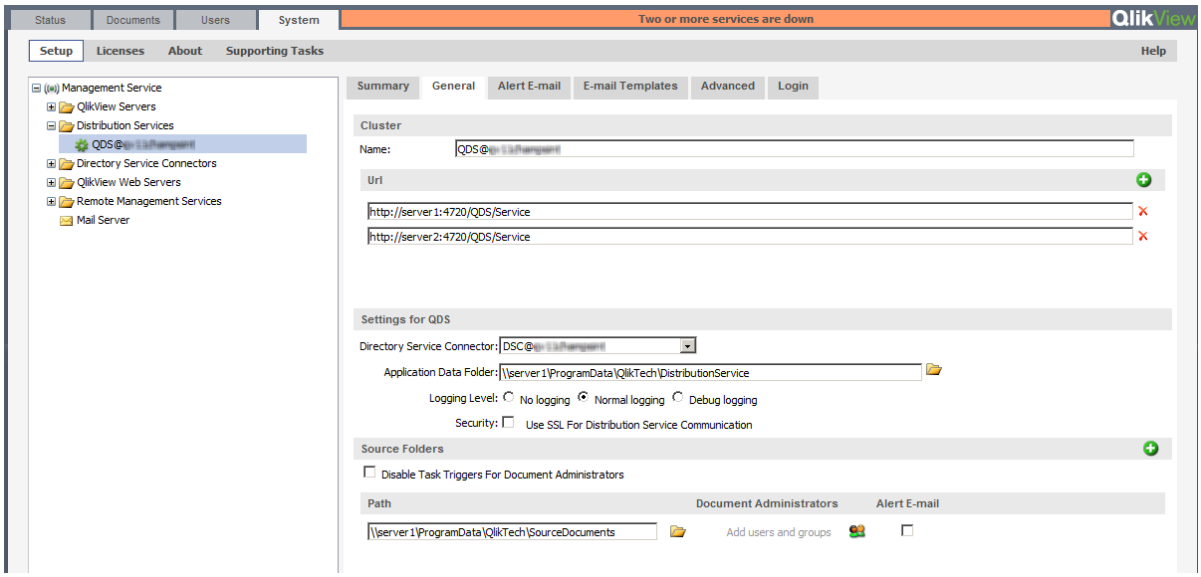
1. Open QMC and register the QlikView Publisher license with the activated cluster nodes.



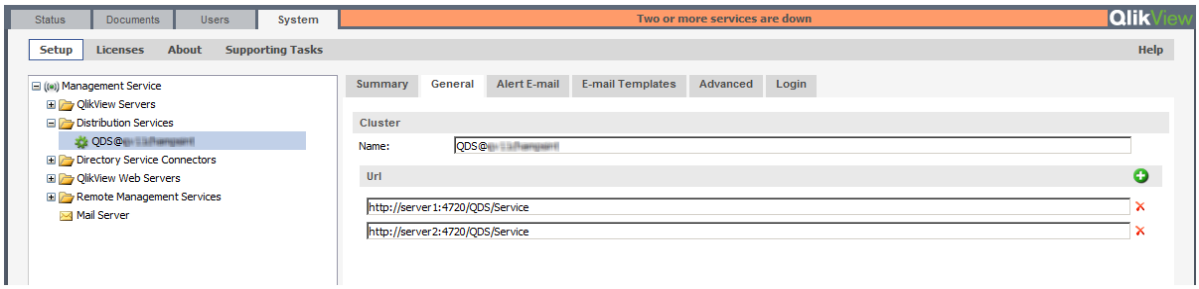

2. On the **System**>**Setup** tab, add the first QDS cluster node under **Distribution Services**.




3. Switch the **Application Data Folder** and the **Source Folders** to the shared device folder paths using UNC syntax.




4. Click **Apply** and restart the QDS manually.
5. Add each additional QDS cluster node in URL format.




6. Click **Apply** and restart the QDS on all nodes manually.

# 18.6 Troubleshooting

If the log message “The network BIOS command limit has been reached” occurs in the debug cluster log, the limit for long-term sessions in the registry has to be increased. Failure to do so may result in tasks not being run.

Increase the following parameters in the registry:

```
HKEY_LOCAL_
MACHINE\SYSTEM\CurrentControlSet\Services\lanmanworkstation\parameters\MaxCmds
```

and

```
HKEY_LOCAL_
MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters\MaxMpxCt
```

**Note!** This issue only occurs on Windows Server 2000, Windows XP, and Windows Server 2003. For more information, see **http://blogs.msdn.com/b/ntdebugging/archive/2007/01/04/desktop-heap-overview.aspx** and **http://support.microsoft.com/kb/810886**.

For QlikView 10 and 11, the settings are available in the `config.xml` file on the server where the QlikView Publisher service is installed:

- Windows Server 2003: `C:\Documents and Settings\All Users\Application Data\QlikTech\DistributionService`
- Windows Server 2008 and later: `C:\ProgramData\QlikTech\DistributionService on Windows xxxx Server`

# 19 OEM

## 19.1 General

The OEM feature prevents abuse of QlikView Servers sold under an Original Equipment Manufacturer (OEM) license and protects the revenue streams of both the OEM products and the full QlikView product. In addition, the feature helps avoid channel conflicts between QlikView OEM partners, QlikView reseller partners, and QlikView direct account managers.

The OEM feature includes the following restrictions:

- A QlikView Server delivered to a customer by an OEM partner cannot run other QlikView applications than the ones delivered by the OEM partner.
- A QlikView application delivered to a customer by an OEM partner cannot run on another QlikView Server than the one delivered by the OEM partner.

## 19.2 Detailed Function Description

The functions of the OEM feature are as follows:

A tag with a key is defined in the QlikView Server License Enabler File (LEF) as `OEM_PRODUCT_ID`. This LEF tag is issued once for each OEM partner and their QlikView Desktop, and QlikView Server licenses with matching tags are delivered for each QlikView Server deployment requiring this feature.

The User Preferences dialog in QlikView Desktop allows an OEM developer to embed a hash key in the `.qvw` file. The hash key, which is based on the `OEM_PRODUCT_ID` key present in the QlikView Desktop license of the OEM partner, is a capitalized 40 character hex string that is stored in the Document Properties and Document metadata. In the dialog, the partner can label all keys generated for the `.qvw` files. The same key can be used for multiple documents belonging to the same customer.

A QlikView Server with the `OEM_PRODUCT_ID` tag in its LEF only permits the publishing of `.qvw` files with a matching key on that QlikView Server. A standard, non-OEM QlikView Server by default opens any `.qvw` file, with the exception of `.qvw` files containing a specific key that some OEM partners are issued with to prevent opening with any other QlikView Server than the one with a matching `OEM_PRODUCT_ID`.

The table below provides a few examples of the results of the OEM functionality.

| | | File | | |
|---|---|---|---|---|
| | | **Normal.qvw** | **OEM 1.qvw** | **OEM 2.qvw** |
| QlikView Server | Normal QlikView Server | File opened | File not opened | File not opened |
| | OEM 1 (No license lease) | File not opened | File opened | File not opened |
| | OEM 2 (No license lease) | File not opened | File not opened | File opened |

In QlikView Desktop, a `.qvw` file containing a `PRODUCT_ID` is opened in user mode.

# 20 DSP Interface

The reason for developing a proprietary Directory Service Provider (DSP) is to have QlikView distribute documents to users in a directory service not supported by default, and to provide group resolution to the web server.

## 20.1 DirectoryServiceProvider

DirectoryServiceProvider is the interface of the class that plugs into the framework. The members of the interface are listed below.

| Member | Description |
|---|---|
| `LogMessage LogMessageEvent { set; get; }` | Directly after construction, this field is instantiated with a delegate that provides crude logging facilities. |
| `string ProviderName { get; }` | A free-form, preferably descriptive, name of the component that is suitable for the end user. |
| `string ProviderType { get; }` | An installation-unique identifier used internally by the framework and related components. The identifiers used by the supplied providers are `AD`, `NT`, `Local`, and `Custom`. |
| `void SetupPath (string _path, string _username, string _password);` | Creates a node that represents the corresponding directory service node on the specified path. Upon failure, an exception is thrown. |
| `IList<string>GetKnownRootPaths ();` | The returned list should contain one or more viable paths for the methods listed here. |
| `void ClearCache ();` | Clears the cache (if any). |
| `string DomainName { get; }` | A “domain name” associated with the path that is set up. It is used as a qualifier to separate nodes from different providers (for example, the shipped Active Directory provider uses `NetBIOSName` as domain name). |
| `IDictionary<string, string> GetSettings ();` | The dictionary of supported settings has the name of the setting as `key` and the name of the type as `value`. |
| `void SetSetting (string _name, string _value);` | The parsing responsibility is obviously put on the provider. |
| `IList<IDSObject> Search (string [] _pattern, eSearchType _type, string _otherattribute);` | Searches for nodes with attributes matching any of the patterns provided. The attributes are specified with the `type` parameter, which can be one or more values from the enumeration. If `type` is “other”, the last parameter specifies the name of the attribute. The search type “legacyid” is used for backwards compatibility. Search should support patterns containing the wildcard sign “*”, which matches zero or more characters of any kind. |

| Member | Description |
|---|---|
| `void Dispose ();` | Called whenever a provider object is released. |
| `IDSObject` | A simple interface for any type of node within the directory service. |
| `string ID { get; }` | Node ID, unique within the instantiated path and consistent over all executions. |
| `string DisplayName { get; }` | Common name of the node in the directory service. |
| `string AccountName { get; }` | Account name associated with the node (if present). |
| `eDSObjectType ObjectType { get; }` | Basic type of the object. |
| `IList<IContainer> MemberOf ();` | A list of all groups that the node is member of. |
| `string GetCustomProperty (string _name);` | Any other property not natively supported by the interface. If not present, null is returned. |
| `string Email { get; }` | The primary e-mail address associated with the node (if any). |

# 21 SNMP

QlikView provides SNMP agents for all services.

**Note!** QlikView supports the iReasoning MIB browser for pulling data from the SNMP agents.

The SNMP setting is off by default, since the implementation is in its initial stages and subject to change. At the time of writing, reading operations from the agents are enabled. The following messages are supported:

- `GetRequest`
- `GetResponse`
- `GetNextRequest`

All services answer the standard SNMP queries (see below).

| Identifier | Query | Description |
|---|---|---|
| 1.3.6.1.2.1.1.1 | sysDescr | Description of service/product.<br>Example:<br>`sysDescr.0:Qlikview Publisher Commandcenterservice version 8.50.600` |
| 1.3.6.1.2.1.1.2 | sysObjectID | Unit type.<br>Example:<br>`sysObjectID.0:iso.org.dod.internet.private.enterprises.qliktech .products.publisher.Distributionservice` |
| 1.3.6.1.2.1.1.3 | sysUpTime | System uptime.<br>Example:<br>`sysUpTime.0:0 hours, 12 minutes, 15 seconds` |
| 1.3.6.1.2.1.1.4 | sysContact | Can be set in the configuration file.<br>Example:<br>`sysContact.0:Unspecified System contact` |
| 1.3.6.1.2.1.1.5 | sysName | Can be set in the configuration file.<br>Example:<br>`sysName.0:Unspecified name` |
| 1.3.6.1.2.1.1.6 | sysLocation | Can be set in the configuration file.<br>Example:<br>`sysLocation.0:Unspecified location` |
| 1.3.6.1.2.1.1.7 | sysService | Constant, 72 means application server.<br>Example:<br>`sysServices.0:72` |

The QlikView Distribution Service can answer additional queries. These are specified in the *MIB File (page 126)*.

Each service has a configuration file, which is stored in the subfolder for the service in the installation folder. For example, the configuration file for the QlikView Distribution Service is `QlikViewdistributionService.exe.config`.

The SNMP settings can be adjusted in the `SNMP SETTINGS` part of the configuration file. SNMP has to be enabled for all services (the default is off).

| Setting | Description |
|---|---|
| EnableSNMP | Enables the SNMP listener. The default value is `false`. |
| SNMPPort | Sets the port to use for the particular Publisher service. See the default settings for each service below. |
| SNMPsysContact | Contact information for the person responsible for the managed node. The default value is `Unspecified System contact`. |
| SNMPsysName | An administratively assigned name for the managed node. By convention, this is the fully qualified domain name of the node. If the name is unknown, the value is a zero-length string. If left empty, it defaults to the current machine name. The default value is `Unspecified name`. |
| SNMPsysLocation | Physical location of the node (for example, “telephone closet, third floor”). The default value is `Unspecified location`. |
| DebugSNMP | Enables the extended debug log for the SNMP listener. The default value is `false`. |

The default port settings for the services are listed below.

| Service | Default Port Setting |
|---|---|
| Management Service | 4781 |
| Directory Service Connector | 4731 |
| Distribution Service | 4721 (default SNMP port) |
| QlikView Server | 161 |
| QlikView Web Server | 4751 |

All ports can be configured. If the services are installed on different machines, they can all run on the same port. The ports change as the implementation moves away from the experimental SNMP range and into the range allotted by QlikTech.

# 21.1 MIB File

A MIB file is included in the QlikView delivery, so that all SNMP managers can interpret the additional responses from the QlikView Distribution Service. Note, however, that the MIB file is subject to change. The file is installed in `\QlikView\Support Tools`. The support tools require a customized installation.

The QlikView Distribution Service can answer the queries listed below, in addition to the ones previously mentioned.

| Identifier | Query |
|---|---|
| 1.3.6.1.4.1.30764.1.2.2.1 | QDSTaskExecuteStatusTable |
| 1.3.6.1.4.1.30764.1.2.2.1.1 | QDSTaskExecuteStatusEntry |
| 1.3.6.1.4.1.30764.1.2.2.1.1.1 | QDSTaskID (task ID number) |

| Identifier | Query |
|---|---|
| 1.3.6.1.4.1.30764.1.2.2.1.1.2 | QDSTaskName (task name) |
| 1.3.6.1.4.1.30764.1.2.2.1.1.3 | QDSTaskExecuteStatus (task status):<br>• Waiting<br>• Running<br>• Aborting<br>• Failed<br>• Warning |
| 1.3.6.1.4.1.30764.1.2.2.1.1.4 | QDSTaskNextExecutionAt (when the task will be executed next) |
| 1.3.6.1.4.1.30764.1.2.2.1.1.5 | QDSTaskLastExecutedAt (when the task was executed last) |
| 1.3.6.1.4.1.30764.1.2.2.1.1.6 | QDSTaskCurrentWork (what the task is currently doing) |
| 1.3.6.1.4.1.30764.1.2.2.1.1.7 | |

For additional information on SNMP, see:

- RFC for SNMP: **http://www.ietf.org/rfc/rfc1157.txt**
- Wikipedia®: **http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol**

# 22 Deploying MSI Packages with Group Policies

**Note!** This chapter is mainly intended for the Internet Explorer plugin.

## 22.1 General

A common problem today is how to deploy applications in a network environment where the users have limited rights, and how to deploy applications for a specific group of users. This section briefly describes how to deploy Microsoft Windows Installer (`.msi`) packages with group policies in an Active Directory environment.

**Note!** Deployment of software with group policies is only supported by workstations running Windows XP Professional, Windows Vista, Windows Server 2003, and later.

The QlikView `.msi` packages require version 2.0 or higher of the Windows Installer service to be installed on the destination workstations.

## 22.2 Deploying the MSI Package

When the `.msi` file has been obtained, it must be placed in a shared folder on the network. Make sure that all users and/or machines that are to install the application have read access to the folder. When the package has been made available to the users and/or machines, the Group policy object that will advertise the installation package can be created.

The package can be advertised to each user or each machine. Use the **User Configuration**>**Software Settings** container to advertise the package per user, and the **Computer Configuration**>**Software Settings** container to advertise per machine. Both containers are located in the Group Policy Object editor.

If the package is advertised per user, it can be either assigned or published. A package that is advertised per machine can only be published.

To publish a package per user means that it is listed (that is, advertised) in the Add programs from your network list in the Add/Remove programs dialog.


*Add/Remove programs dialog*

Each user must click the **Add** button to complete the installation.

To publish a package per machine means that the package is installed and accessible to all users on that machine the next time the machine is rebooted.

An advertised package that is assigned is also listed in the **Add programs from your network** list and can be added from there. This option also offers a few more ways to activate the installation package:

- Shortcuts (if the installation package adds any) on the desktop and/or Start Menu: The shortcuts are added and the installation package can be executed by clicking the appropriate shortcut.
- File association: The installation program is executed when the user tries to open a file that is associated with the advertised application.

There are a few more ways to execute the installation when it is advertised as assigned, but they are not applicable to any QlikView installations and therefore beyond the scope of this document.

**Note!** The Internet Explorer plugin installation package does not add any shortcuts or file associations. It is therefore not recommended to advertise QlikView installation packages with the assign option.

## Advertising

To advertise means that the administrator gives the installation package permission to execute on an account with locked down permissions.

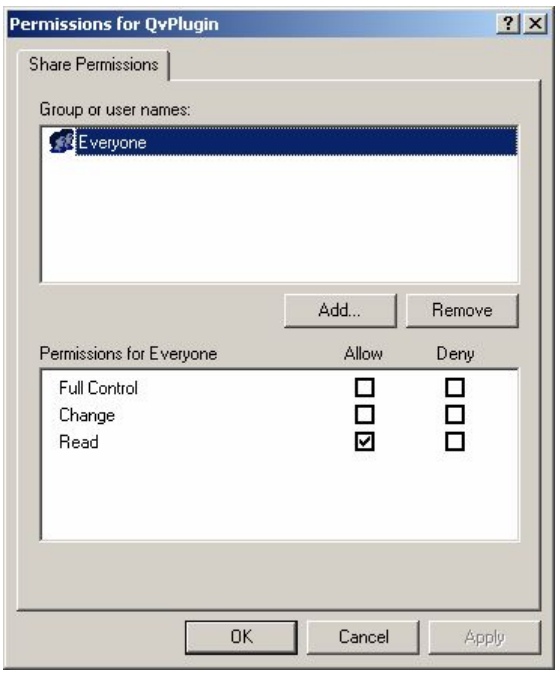
When the package is advertised, there are so called “entry points” loaded onto the destination system. Entry points are typically shortcuts, file associations, listing in the Add/Remove Programs dialog, and so on.

# 22.3 Step-by-step Guide

This section provides a step-by-step guide for creating a group policy for advertising of the QlikView Internet Explorer plugin `.msi` package on a number of machines in the Active Directory.

Proceed as follows to create a group policy:

1. Browse to the folder containing the `.msi` package. Share the folder with the network users with permission to install the package.




*Sharing the folder*

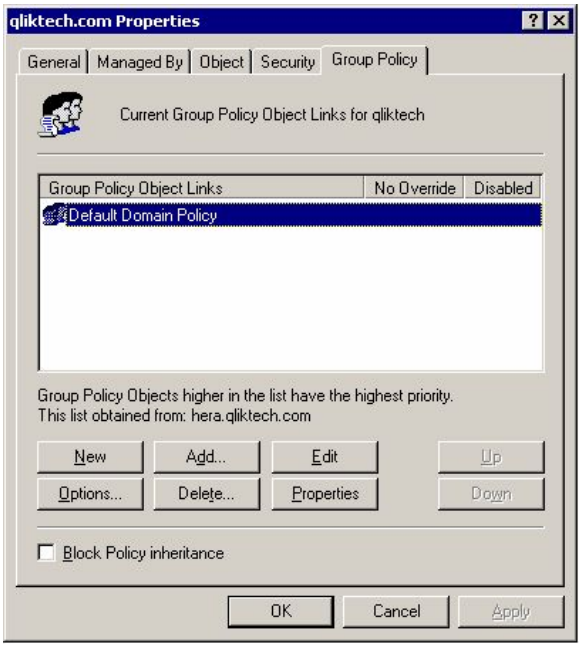
2. Open **Active Directory Users and Computers** and highlight the **Organizational Unit (OU)** where the package is to be deployed.


*Highlighting the Organizational Unit where to deploy the package*

3. Right-click and select **Properties**.

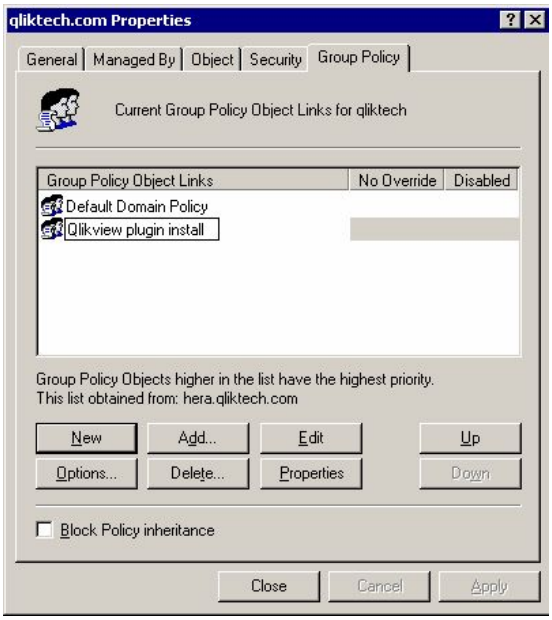

*Selecting Properties*

4. Go to the **Group Policy** tab, click **New**, and give the group policy object an appropriate name.



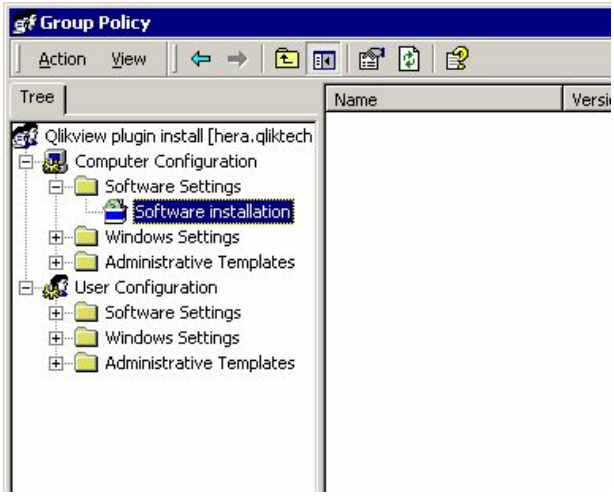

*Providing a name*

5. Highlight the new group policy object and click **Edit**.



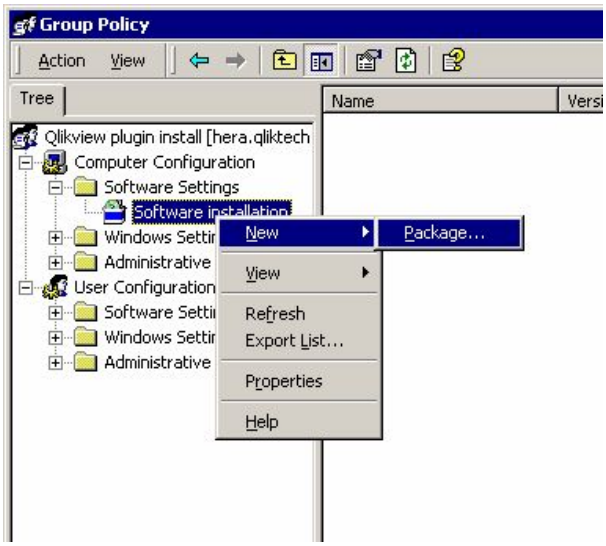

*Highlighting the new group policy object*

6. Expand **Computer Configuration**>**Software Settings** or **User Configuration**>**Software Settings**, depending on how the package is to be deployed. In this case, **Computer Configuration**>**Software Settings** is selected.




*Selecting Software Settings*

7. Right-click **Software installation** and select **New**>**Package...** A pop-up window, asking where to locate the installation package, is displayed.

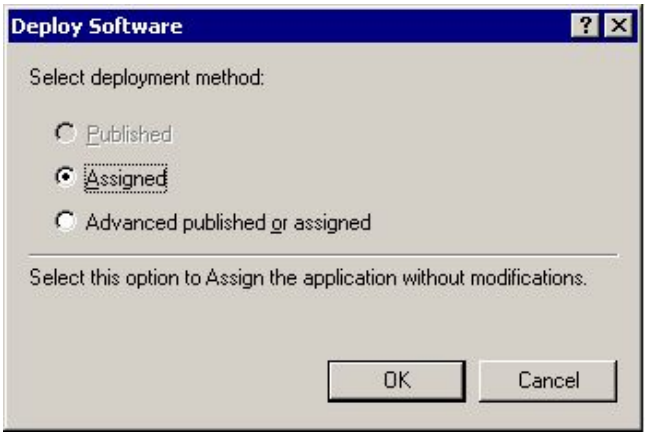



*Creating a new package*

8. Browse to the installation package (in this case, `QvPluginSetup.msi`), select it, and click **Open**.

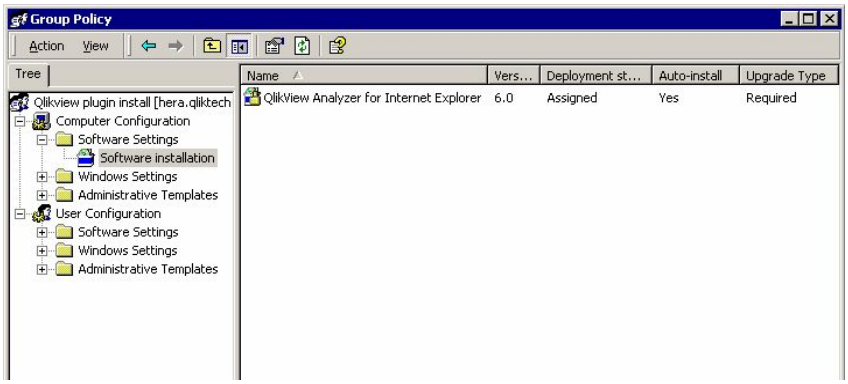

*Opening the installation package*

9. Select the deployment method **Assigned** and click **OK**. Since the installation is to be applied to the **Computer Configuration**, only the **Assigned** deployment method can be used.




*Selecting deployment method*

10. The deployment rule is now ready for use. All machines in the OU get this deployment automatically. What actually happens is that when a machine is rebooted, the installation program is executed, so that any user that logs on to a machine in that OU can run the installed program. The rule can be applied to many different OUs.




*Deployment rule is ready for use*

# 23 Certificate Trust

QlikView 11 Server uses certificates for authentication and authorization. A certificate provides trust between servers (that is, machines).

This chapter describes how to deploy certificates on multiple servers.

## 23.1 Architecture

Certificates are used in a QlikView installation to authenticate and authorize communication between services that reside on multiple servers. Configuring certificates in a multiple server deployment within QlikView removes the dependency on a QlikView Administration Group for the establishment of trust between the QlikView services. It also allows the use of certificates to build a trust domain between QlikView services that are located in different domains without having to share an Active Directory (AD) or other user directories.

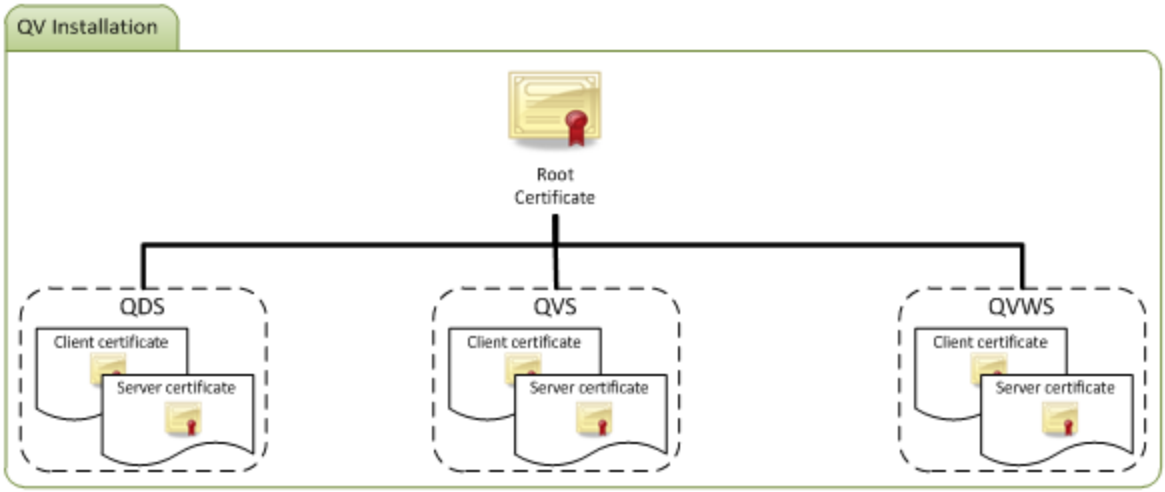
**Note!** The configuration steps described in this chapter only provide a trust domain between the Qlikview services. The use of SSL and certificates for securing end-user communication has to be configured separately.

The architecture is based on the QlikView Management Service (QMS) acting as the certificate manager or Certificate Authority (CA). The QMS can create and distribute certificates to all services in the QlikView installation.

QMS is therefore an important part of the security solution and has to be managed from a secure location to keep the certificate solution secure.


The root certificate for the installation is stored on the QMS server. All servers with QlikView services that are to participate in the installation receive certificates signed using the root certificate when added to the

QMS. The QMS (that is, the CA) issues digital certificates that contain keys and the identity of the owner. The private key is not made publicly available – it is kept secret by the QlikView services. The certificate enables the QMS to validate the authenticity of the service. This means that the QMS is responsible for saying “yes, this service deployed on this server is a service in my installation”.




After the servers have received certificates, the communication between the QlikView services is encrypted using HTTPS (SSL encryption). The certificates only secure the communication between the services on the servers. The certificates do not secure the communication with the end user (that is, the certificates are not used for QlikView plugin, client, or web server communication with the QVS).

# 23.2 Requirements

## General

The following requirements must be fulfilled for the certificate trust to function properly:

- Certificate trust cannot be partially implemented. It is either used by all services in the QlikView installation or not at all.
- Certificate trust is only supported by Windows Server 2008 and later.
- If running QlikView 9/10 Server, upgrade to QlikView 11 Server.
- If it is an initial install of QlikView 11 Server, install and configure the QlikView services without any modification. Prior to configuring the use of certificates, start and stop the services on the servers (that is, machines) where the QlikView services are deployed.
- Section Access management must not be configured in environments where certificate trust is configured.

In addition, the technical requirements described in the following sections also have to be fulfilled.

## Communication Ports

A number of ports are used for service communication using certificate trust. For all services, except for communication with the QVS, the web services protocol SOAP over SSL is used on the specified port. In case of the QVS, QVPX is used as the protocol over SSL.

When using certificate trust for service communication, the servers require that the ports listed in the following table can be opened and used for communication. If any QlikView communication passes through a network firewall, the ports in the firewall must be opened and configured for the QlikView services.

| Service | Port |
|---|---|
| QlikView Distribution Service (QDS) (Publisher) | 4720/TCP |

| Service | Port |
|---|---|
| Directory Service Connector (DSC) | 4730/TCP |
| QlikView Management Service (QMS) | 4780/TCP |
| QlikView Webserver (QVWS)/IIS configuration | 4750/TCP |
| QVS configuration | 4749/TCP |
| QVP communication | 4747/TCP |
| QMS (EDX calls) | 4799/TCP |

Firewall configuration changes might be necessary, depending on the location of the QlikView servers within the resulting network and the routing of the QVS communication.

### Access

To install the distributed certificates for the respective services, physical access to the console or remote access to the console (for example, using remote desktop functionality) is needed.

## 23.3 Installation

Only install the QlikView services (components) needed on each server. Do not perform a full install on all servers – use "custom install" and select only the services that will be active and executing on each server in the QlikView configuration. To simplify the procedure, it is recommended to have the same Windows Administrator on all servers in the QlikView configuration.

### Enabling Certificate Service Authentication

Proceed as follows to enable certificate service authentication for DSC, QVWS, QMC, QDS, and QVS:

1. Stop the QMS service.
2. Open the `C:\Program Files\QlikView\Management Service\QVManagementService.exe.config` file in Notepad.
3. Change the "UseWinAuthentication" entry from "true" to "false".
4. Save the file.
5. Right-click the file and select **Run as administrator**.
6. Start the QMS service.

At this point, you can check if the certificates are properly set on the server that executes the QMS service by running the Microsoft Management Console (MMC) from the Start menu. See Using Microsoft Management Console for details.

### Configuring Certificates

Proceed as follows to configure the certificates for the remaining servers:

1. Stop the DSC, QDS, QVWS, and IIS services on the servers where they are located.
2. Open the `<service>.exe.config` file for each service in Notepad.

| Service | Default Path |
|---|---|
| DSC | `C:\Program Files\QlikView\Directory ServiceConnector\QVDirectoryServiceConnector.exe.config` |

| Service | Default Path |
|---|---|
| QDS | `C:\Program Files\QlikView\Distribution Service\QVDistributionService.exe.config` |
| QVWS | `C:\Program Files\QlikView\Server\Web Server\ QVWebServer.exe.config` |
| IIS | `C:\Program Files\QlikView\Server\Web Server Settings\QVWebServerSettingsService.exe.config` `C:\Program Files\QlikView\Server\QlikViewClients\QlikViewAjax\web.config` |

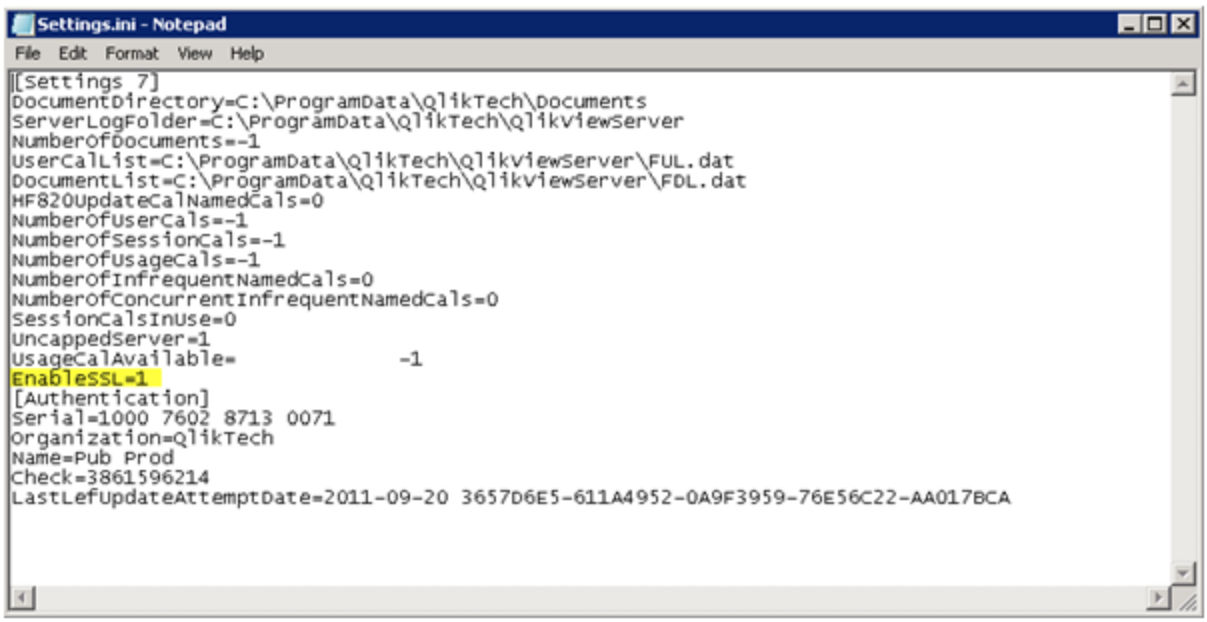
3. Change the "UseWinAuthentication" entry from "true" to "false" in each file.
4. Save the files.
5. Right-click each file and select **Run as administrator**.
6. Start the DSC, QDS, QVWS, and IIS services on the servers where they are located.

Certificate trust with IIS and QlikView 11 Server is configured using port 4750 (that is, the same port as the QVWS uses). The .aspx page, which required port 80 or 443 in QlikView 10 Server, is longer used. However, the certificate trust used to enable HTTPS access for users of the web server remains unchanged.

# Editing the Settings.ini File

Proceed as follows to edit the Settings.ini file for the QVS service:

1. Stop the QVS service.
2. Open the `C:\ProgramData\QlikTech\QlikViewServer\Settings.ini` file in Notepad.
3. Add `EnableSSL=1` in the `[Settings 7]` section.

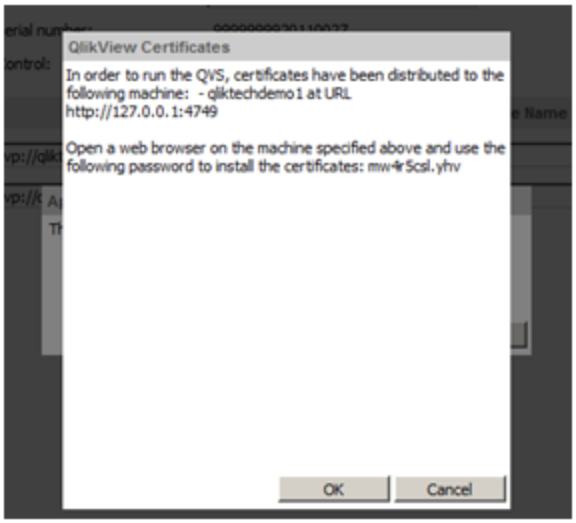



4. Save the file.
5. Right-click the file and select **Run as administrator**.
6. Start the QVS service.

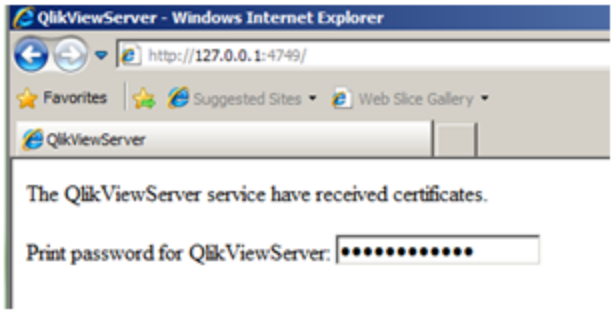
# Adding Services to Issue the Certificates

Proceed as follows to add the services to issue the certificates:

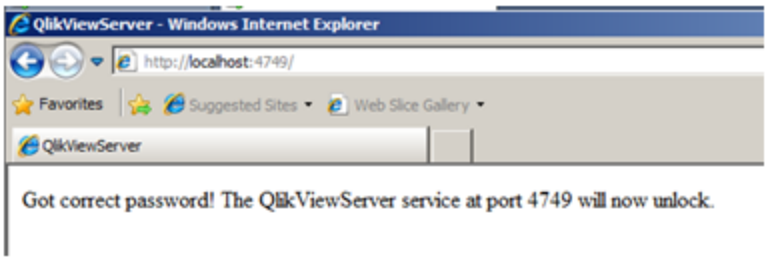
1. Open the QlikView Management Console (QMC).
2. Add each service as a new service and then delete the existing service.
3. When adding a service, a popup window appears.




4. Access the server where the new service resides, either physically or by using a remote desktop connection. Then open a web browser and enter the URL and port provided by the QMC popup window.
5. On the resulting web page, enter the password provided the QMC popup window.




6. If successful, you receive the message below.




At this point, you can check to see if the certificates are properly set up on the servers that execute the additional QlikView services by running the MMC from the Start menu. See Using Microsoft Management Console.

## Using Multiple Services on a Single Server

A certificate provides “trust” between servers (that is, machines). If you have installed multiple QlikView services on the same server, proceed as follows:

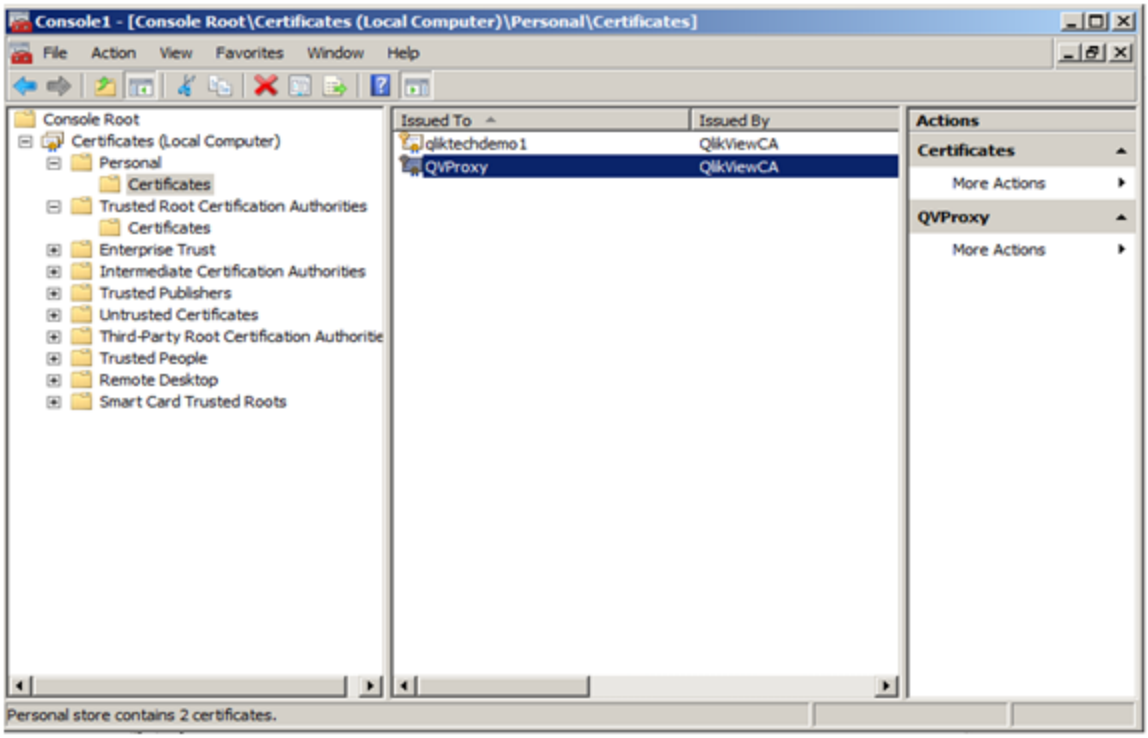
1. Stop the additional QlikView services.
2. Open the `.config` file for each service.
3. Change the “UseWinAuthentication” entry from “true” to “false” in each file.

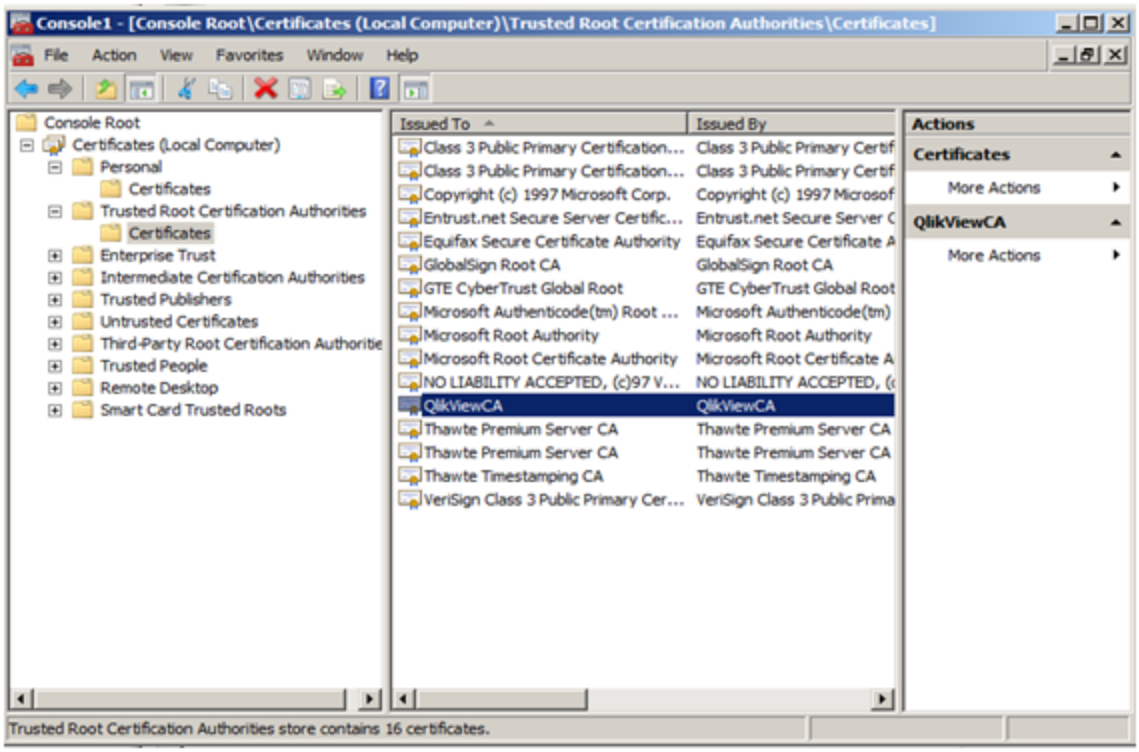
4. Save the files.
5. Start the service. No additional popup window is displayed and you do not have to enter a password for any additional QlikView services.

**Note!** All TCP ports (4720, 4730, 4747, 4749, 4750, 4780, and 4799) must be configured as “open”.

## 23.4 Using Microsoft Management Console

Certificates can be visually confirmed in the MMC with the certificate snap-in added. The QlikView certificates are located in the **Personal>Certificates** and **Trusted Root Certification Authorities>Certificates** folders:






The figures above show properly installed certificates in a QlikView 11 Server configuration. Within the MMC, all QlikView services on servers have certificates deployed as shown in the figures.

The uninstaller does not remove the certificates. This means the certificates have to be deleted manually, if needed.

# 24 QlikView Server Extensions

## 24.1 Adding Extensions to QlikView Server

To run QlikView Extensions on a QlikView Server, the contents of the `Extensions` folder have to be copied from `%UserProfile%\AppData\Local\QlikTech\QlikView\Extensions\Objects` to the `%ProgramData%\QlikTech\QlikViewServer\Extensions\Objects` folder on the server.

If the path to the extensions is changed (for example, to a common place for all servers in a cluster), that path must be used instead. Note that the path set corresponds to `%UserProfile%\AppData\Local\QlikTech\QlikView\Extensions` (that is, it does not include `\Objects`).

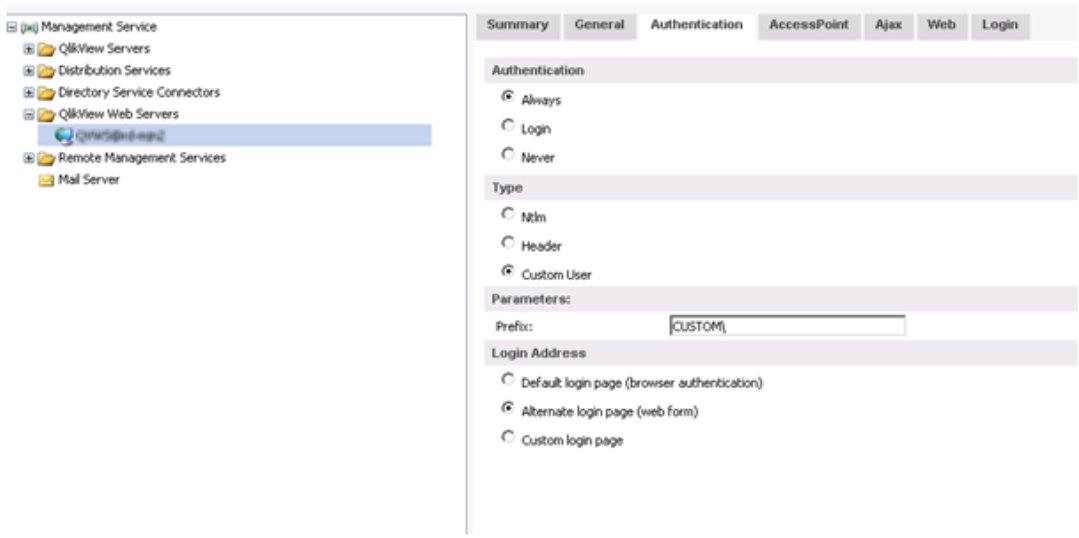
# 25 Configuring Microsoft IIS for Custom Users

When using Microsoft IIS as web server for Custom Users, configuration is needed.

Proceed as follows to configure IIS for Custom Users:

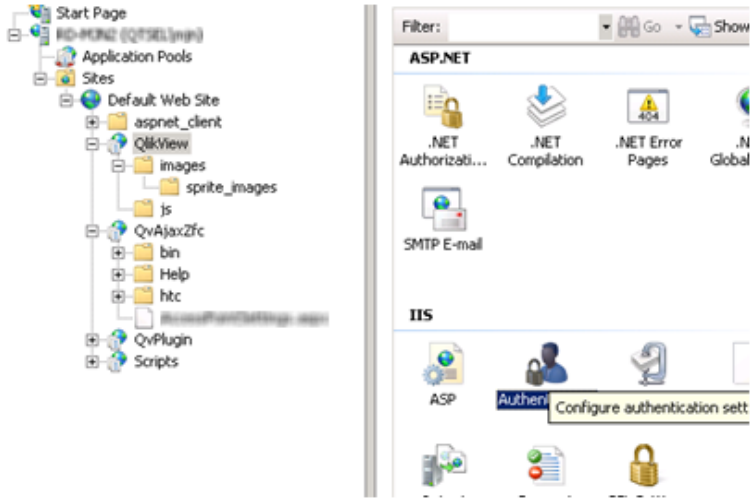
1. In QlikView Management Console, change the parameters on the **System**>**Setup**>**Authentication** tab in accordance to the following:

| | |
|---|---|
| **Authentication** | Always |
| **Type** | Custom User |
| **Parameters** | CUSTOM\ |
| **Login Address** | Alternate login page (web form) |



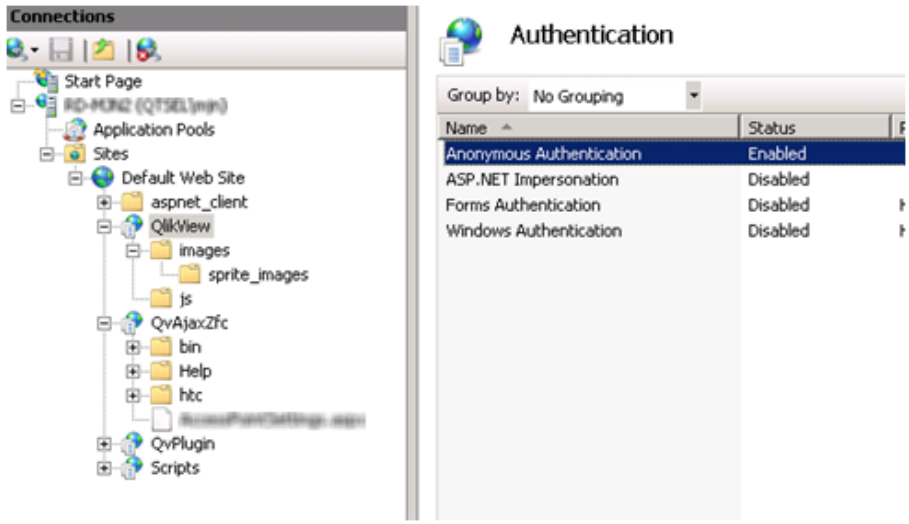

*Authentication tab*

2. Select the `QlikView` virtual folder and then **Authentication**.



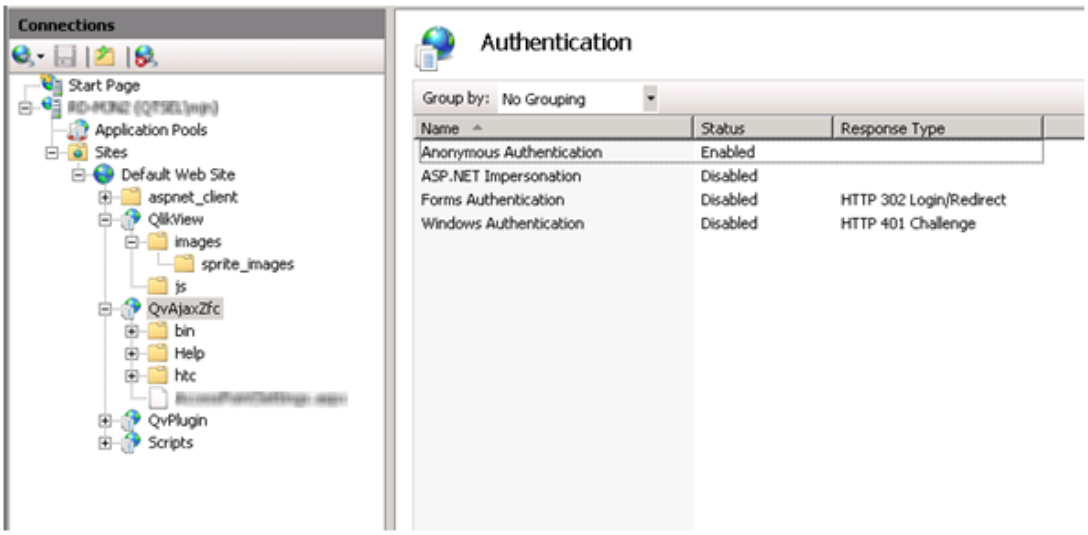

*Selecting Authentication*

3. Disable **Windows Authentication** and enable **Anonymous Authentication**.



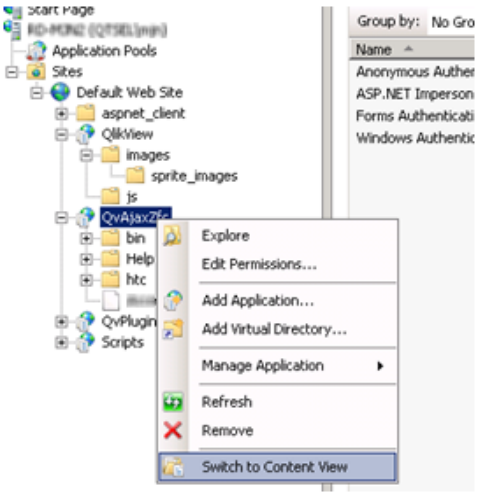

*Enabling Anonymous Authentication for the QlikView virtual folder*

4. Select the `QvAjaxZfc` folder and then **Authentication**.
5. Disable **Windows Authentication** and enable **Anonymous Authentication**.




*Enabling Anonymous Authentication for the QvAjaxZfc folder*

6. Right-click `QvAjaxZfc` and select **Switch to Content View**.




*Selecting Switch to Content View*

7. The configuration of IIS for the Custom User is complete.

# 26 Triggering EDX Enabled Tasks

To start tasks that have an external event as trigger, the QlikView Management Service API (QMS API) must be used. The user making the request calls must be a member of the QlikView Administrators local group or the QlikView EDX local group. The QlikView Administrators group is set up during the installation of QlikView Server, but the QlikView EDX group must be created manually in **Computer Management**. Members of the QlikView EDX group only have the right to trigger EDX-enabled tasks.

The method to use has the following signature:

```
TriggerEDXTaskResult TriggerEDXTask(Guid guid, string taskNameOrId,
                                    string password, string variableName,
                                    List<string> variableValues)
```

| Parameter | Purpose |
|---|---|
| `guid` | ID of the QlikView Distribution Service (QDS) where the task is defined. |
| `taskNameOrId` | Task name or ID of the task in string format. |
| `password` | Password (if required by the task). |
| `variableName` | Variable name (if required by the task). |
| `variableValues` | List of values for the variable. |

The returned result contains information on whether the task was successfully started or not.

The example below shows how to trigger a task and wait until it has finished or until a certain amount of time has passed.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading;
using QMSAPI;

class Program
{
        static void Main(string[] args)
        {
                try
                {
                        // create a QMS API client
                        IQMS apiClient = new QMSClient();

                        // retrieve a time limited service key
                        ServiceKeyClientMessageInspector.ServiceKey =
                        apiClient.GetTimeLimitedServiceKey();

                        //Get a Distribution Service.
                        ServiceInfo qdsService =
                        apiClient.GetServices(ServiceTypes.QlikViewDistributionService).FirstOrDefault();

                        if (qdsService != null)
                        {
                                //Trigger the task
                                TriggerEDXTaskResult result =
                                apiClient.TriggerEDXTask(qdsService.ID, "PauseEDX", "edx", "", new List<string>());

                                EDXStatus executionStatus = null;

                                //Wait until the task is completed or 60 seconds has passed.
                                SpinWait.SpinUntil(() =>
                                        {
                                                System.Threading.Thread.Sleep(1000);
                                                Console.WriteLine("Checking the task...");

                                                //Get the current state of the task.
                                                executionStatus =
                                                apiClient.GetEDXTaskStatus(qdsService.ID, result.ExecId);

                                                //Return true if the task has completed.
                                                return executionStatus !=
                                                null && executionStatus.TaskStatus == TaskStatusValue.Completed;
                                        }, 60 * 1000);

                                        //Write the result
                                        if (executionStatus != null)
                                                Console.WriteLine(executionStatus.TaskStatus);
                                        else
                                                Console.WriteLine("Failed to get execution status.");

                        }
                }
                catch (Exception ex)
                {
                        Console.WriteLine("An exception occurred: " + ex.Message);
                }
                // wait for user to press any key
                Console.ReadLine();
        }
}
```

The example comes from the QMS API documentation, which is installed as part of the QlikView Management Console (QMC). It contains detailed information on the available methods and how to get started with the QMS API.